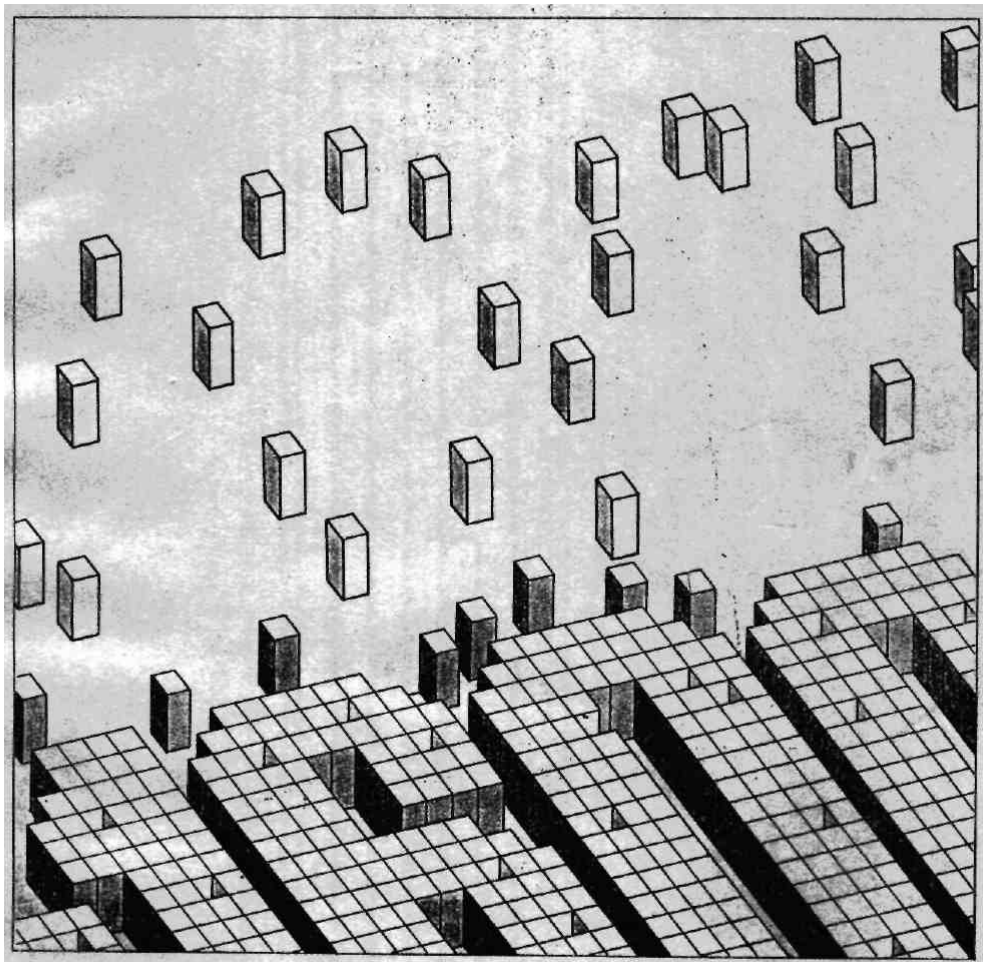


TABLE DES MATIERES

INTRODUCTION
UTILISATION DE CE MODE D'EMPLOI

| PREPARATION A L'UTILISATION 1

1 Description du PC-1600 3
Notice de déballage 3
Vue d'ensemble du matériel 3
Le clavier du PC-1600 6
Vue d'ensemble du logiciel 8

|| FONCTIONNEMENT 9

2 Alimentation du PC-1600 11
Mise en place des piles 11
Témoin d'épuisement des piles 12
Branchement sur le secteur 13

>

3 Mise sous et hors tension 15
Mise sous tension 15
Mise hors tension 19
Réinitialisation de l'ordinateur 20

4 L'affichage du PC-1600 25
Réglage du contraste d'affichage 25
Symboles de la ligne d'état 26
Mode d'exploitation 27
Fonctions des touches d'édition 28
Réglage de l'heure et de la date 31

5 Calculs sur le PC-1600 35
Spécification du mode 35
Fonctions des touches 35
Exemples de calculs 36
Calculs en série 37
Fonction de rappel 38
Codes d'erreur 39
Opérations des fonctions BASIC 40

UTILISATION DES INTERFACES ET DES EQUIPEMENTS E/S EN OPTION 41

6 Elargissement du PC-1600 43
Vue d'ensemble du système 43
Module d'expansion de RAM 45
Interfaces série 49
Port série RS-232C 49
Port série optique 53
Port d'entrée analogique 56
Imprimante avec interface à cassette 58
Utilisation d'un enregistreur à cassette 61

Unité de disque 65

7 Utilisation des périphériques du PC-1500 71

IV SECTION DE REFERENCE DU LANGAGE BASIC 73

8 Concept de base de programmation 75
Introduction des instructions en BASIC 75
Exécution d'un programme 76
Affectation de la mémoire 78
Fichiers en exécution automatique 80

9 Modes de fonctionnement en langage BASIC 83
Manipulation du clavier 83
Modes d'affichages 87
Mode d'édition 89
Mode de réserve 91

10 Représentation des données 97
Types de données 97
Constantes 98
Variables 100
Expressions et opérateurs 105

11 Fichiers 111
Descripteurs de fichiers 111
Stockage et extraction du fichier 113
Protection du fichier 115
Création d'un fichier 116
Accès au fichier 118
Mise à jour du fichier 119

12 Accès aux ports série 121
Spécification du port.....121
Paramètres de communication 122
Sortie à un port série.....122
Entrée par un port série 124

13 Mise au point..... 125
Erreurs de syntaxe.....125
Mode d'analyse 125
Programmes de traitement d'erreurs.....126

14 Dictionnaire des commandes BASIC 127

V ANNEXES 343

A. REMPLACEMENT DES PILES.....345
B. REMPLACEMENT DES MODULES DE RAM 347
C. TABLEAU DES CODES DE CARACTERES 349
D. TOPOGRAPHIE MEMOIRE.....351
E. PROGRAMMES LANGAGE MACHINE.....354
F. CODES ERROR POUR LE PC-1600 356
G. LISTE DES COMMANDES BASIC 363
H. COMPATIBILITE AVEC LE PC-1500 ET SES PERIPHERIQUES.....368
I. ENTRETIEN ET PANNES 372
J. SPECIFICATIONS 375

INTRODUCTION

Nous vous remercions d'avoir choisi l'ordinateur de poche SHARP PC-1600. Le PC-1600 vous offre le maximum dans la portabilité et la souplesse d'utilisation. Le PC-1600, seul, tient dans la paume de votre main, mais est doté de la même puissance de calcul qu'un ordinateur de taille beaucoup plus grande que lui. Connecté avec une gamme complète d'équipements en option, spécialement conçus, qui sont eux aussi de format réduit tout en offrant les fonctions et les caractéristiques des périphériques standards, votre PC-1600 se transforme alors en un système compact, pour répondre à une grande variété de besoins.

Voici quelques-unes des caractéristiques standards qui font de votre PC-1600 le choix parfait, aussi bien pour les utilisateurs expérimentés que pour les débutants ou presque débutants.

- •Microprocesseur puissant à 8 bit avec MOS complémentaire, équivalent au Z-80 A
- •Mémoire vive standard (RAM) de 16 k-octets, extensible jusqu'à 80 K
- •Affichage à cristaux liquides de 4 lignes et 26 caractères par lignes avec une matrice de caractères de 5 x 7
- •Graphiques de 156 x 32 points, tous les points étant adressables
- •Horloge en temps réel avec fonctions de réveil-matin et d'alarme

Votre PC-1600 fonctionne avec une version spécialement adaptée du langage BASIC qui est dans la mémoire de l'ordinateur et vous permet d'écrire vos propres programmes facilement et rapidement.

Le PC-1600 est alimenté par piles pour favoriser la mobilité et dispose également d'un adaptateur secteur pour son utilisation à la maison ou au bureau.

Vous pouvez également connecter directement le PC-1600 sur des équipements E/S externes via les trois ports incorporés des interfaces: les ports série E/S optiques et RS-232C et le port d'entrée analogique.

Le PC-1600 peut être utilisé avec les accessoires optionnels suivants:

- •Le module de programme CE-1600M qui procure une extension de la RAM de 32 K avec système de sauvegarde sur pile pour la mémorisation des programmes, l'extension de l'espace alloué à la mémoire ou la création d'un disque de RAM.
- •L'imprimante CE-1600P avec interface à cassette pour une impression en quatre couleurs des textes et graphiques et qui permet la connexion d'un enregistreur à cassette pour stocker les programmes et données sur bandes.
- •L'unité de disque CE-1600F pour disques de poche à double face de 2 W pouvant enregistrer 61 K par face.

Le PC-1600 est également compatible avec la gamme des options du SHARP PC-1500 qui comprend une imprimante, un enregistreur à cassette, des modules de RAM, des interfaces et des programmes BASIC.

UTILISATION DE CE MODE D'EMPLOI

Pour favoriser l'accès aux informations concernant votre ordinateur de poche SHARP PC-1600, nous avons découpé ce mode d'emploi en cinq parties :

La partie I, Préparation à l'utilisation, donne une description d'ensemble du PC-1600, pour vous familiariser avec ses éléments composants et ses caractéristiques. Il est indispensable de lire cette partie.

La partie II, Fonctionnement, vous assiste depuis la préparation initiale du PC-1600, y compris l'installation des piles, la mise sous ou hors tension de l'ordinateur et le réglage de l'heure et de la date de l'horloge en temps réel incorporée. Elle vous montre aussi comment utiliser les touches d'édition quand vous introduisez des données et conclut sur un chapitre de l'utilisation du PC-1600 en tant que calculatrice.

La partie III, Utilisation des interfaces et des équipements E/S en option, décrit l'utilisation des trois ports d'interface et des logements des modules de mémoire du PC-1600. Elle vous procure aussi les informations sur l'utilisation du PC-1600 avec son imprimante, son unité de disque et son enregistreur à cassette en option.

La partie IV, Section de référence du langage BASIC, décrit le BASIC, le langage de programmation incorporé qui gère le PC-1600, vous permet d'écrire vos propres programmes et commande les équipements en option. Cette section introduit simplement les concepts généraux du BASIC et son utilisation sur le PC-1600, mais n'a pas pour but de vous enseigner les "bases du BASIC". Nous vous suggérons de vous reporter à l'un des excellents livres disponibles sur ce langage, à la portée de tous. Cette partie comprend un dictionnaire complet des commandes et instructions du BASIC qui sont utilisées dans le PC-1600.

La partie V, Annexes, comprend les informations sur le changement des piles et des modules de RAM. Elle procure également diverses informations techniques et le résumé des codes des erreurs, les spécifications, les topographies mémoire et une liste des commandes BASIC.

Nous vous conseillons de vous familiariser avec ce manuel avant d'utiliser votre ordinateur, afin d'éviter des problèmes ultérieurs. La partie I, Préparation pour l'utilisation, est particulièrement importante car elle présente toutes les caractéristiques du PC-1600. Les autres parties doivent être examinées, puis utilisées au fur
et à mesure en tant que référence.

PARTIE I

PREPARATION A L'UTILISATION

Nous commençons cette partie par une description d'ensemble du PC-1600, après que vous ayez déballé l'ordinateur et ses accessoires et vérifié que le contenu du carton est bien complet.

Puis, les caractéristiques propres du PC-1600 sont alors introduites en accentuant sur le clavier et les fonctions des touches.

Enfin, la dernière section se termine par une vue d'ensemble du logiciel d'exploitation et du rôle du langage BASIC du PC-1600.

1 Description du PC-1600

Nous savons que vous désirez vous servir de votre PC-1600 immédiatement. Afin de l'utiliser d'une manière correcte et d'éviter des problèmes ultérieurs, prenez quelques minutes pour parcourir les pages suivantes et regarder votre PC-1600, avant de passer aux autres sections de ce manuel.

Notice de déballage

Après avoir déballé votre PC-1600 et ses accessoires, contrôlez le contenu avec la liste suivante pour vous assurer que votre lot est bien complet.

Si un élément était manquant ou endommagé, contactez immédiatement votre concessionnaire SHARP.

Le lot de votre PC-1600 comprend:

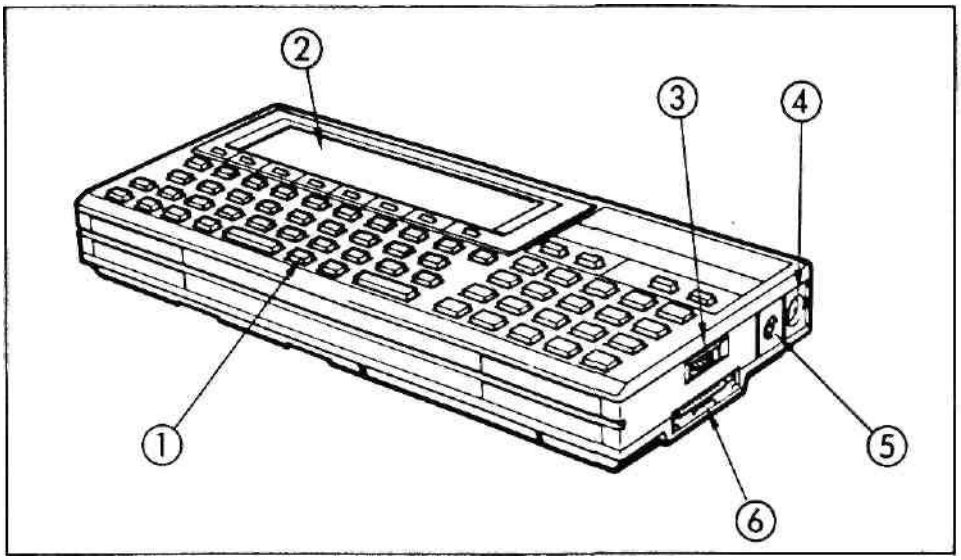
- L'ordinateur de poche SHARP PC-1600
- Un étui souple
- Deux grilles de clavier
- Quatre piles AA
- Ce manuel d'utilisation

Après avoir terminé cette vérification, remettez les matériaux d'emballage dans leur paquet et conservez-les en vue d'une utilisation ultérieure. Vous pouvez avoir besoin d'emballer à nouveau votre PC-1600 pour l'expédier ou pour le ranger pendant une longue période.

Vue d'ensemble du matériel

Le PC-1600 se distingue par son clavier QWERTY, similaire à celui des machines à écrire habituelles, son affichage à cristaux liquides (LCD) avec contraste réglable, ses deux logements de modules pour l'extension de mémoire et ses trois connecteurs d'interface pour raccorder les divers équipements en option.

Les pages suivantes décrivent les composants propres au PC-1600, pour vous familiariser avec leurs emplacements et leurs fonctions.



Vue frontale du PC-1600

1Le clavier Avec sa configuration standard du type machine à écrire, le clavier dispose de 69 touches et offre un certain nombre de touches de fonctions programmables à usage spécial.

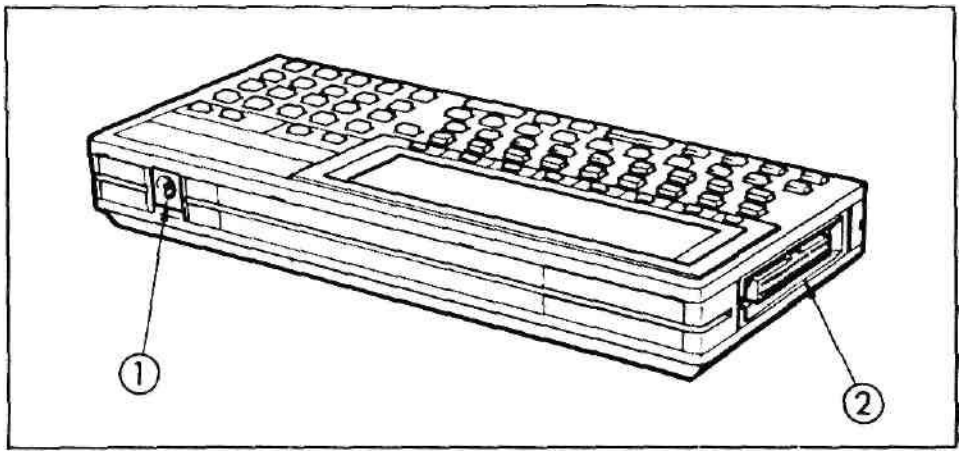
2L'écran LCD L'affichage sur l'écran du PC-1600 est de quatre lignes de 26 caractères par ligne et son contraste est réglable pour ne pas fatiguer la vue.

3Molette de contraste LCD Lorsque la position d'utilisation de l'ordinateur ou les conditions de luminosité changent, cette molette permet d'éclaircir ou d'assombrir l'écran, pour faciliter la lecture.

4Port série E/S optique Utiliser ce connecteur pour brancher le câble à fibres optiques d'un équipement extérieur pour des communications séquentielles ultra-rapides et antiparasitées.

5Port d'entrée analogique Ce port permet au PC-1600 de recevoir des données en provenance des sources analogiques telles que les capteurs et les lecteurs des codes à barres.

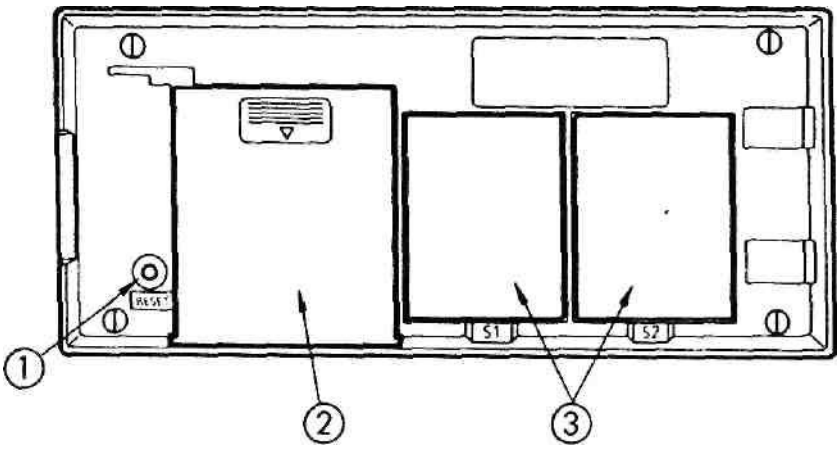
6 Port RS-232C Pour connecter un câble à ce port, en provenance d'un autre ordinateur, d'un modem ou d'une imprimante, pour envoyer ou recevoir des données en série.



Vue arrière du PC-1600

1 Jack d'adaptateur CA Lors de l'utilisation du PC-1600 à l'intérieur ou pour économiser les piles, cet adaptateur (EA-160 ou EA-150) est branché sur ce connecteur afin que l'ordinateur soit alimenté à partir d'une prise électrique murale normalisée.

2 Bus de système Le connecteur de bus procure une liaison directe avec le système de l'ordinateur pour raccorder les interfaces et l'imprimante en option et contrôler la transmission des données entre celles-ci et l'ordinateur.



Vue arrière du PC-1600

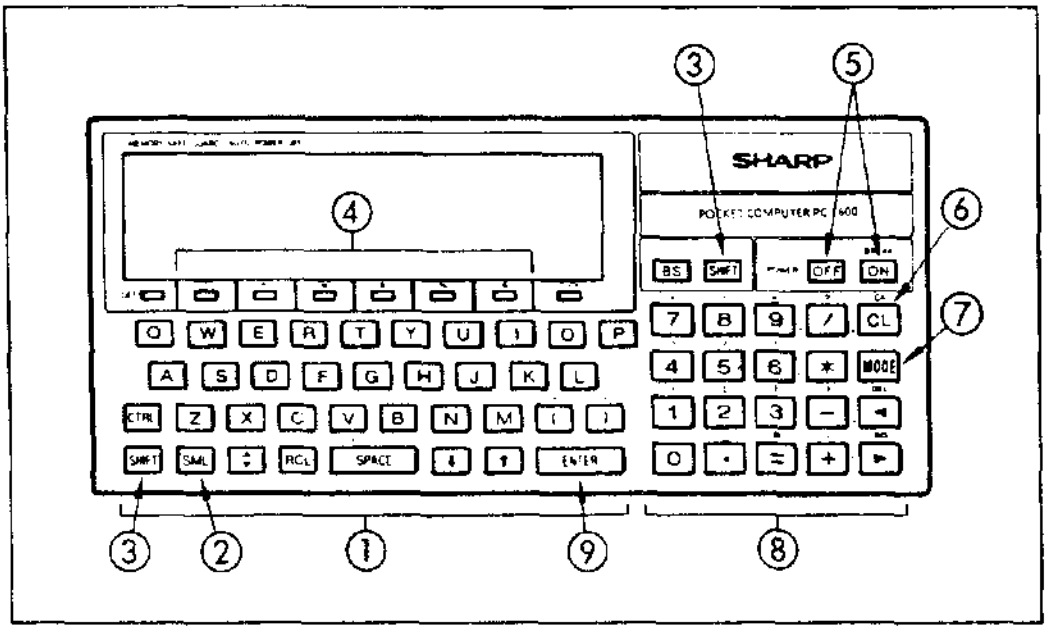
1 Touche de réinitialisation Si le PC-1600 se bloque pour une raison quelconque, l'écran et le clavier seront inopérables. Vous ne pourrez pas utiliser les touches **OFF** et **ON** pour relancer l'ordinateur. La pression de la touche de réinitialisation vide la mémoire du PC-1600 et remet l'ordinateur à la condition qui est juste après la mise sous tension. Utilisez cette touche avec la plus grande prudence car vous pouvez perdre vos programmes et données. Reporte z-vous à la partie 3 -Réinitialisation de l'ordinateur - avant de l'utiliser.

2 Logement des piles Quand le PC-1600 n'est pas raccordé à l'alimentation secteur par l'adaptateur en option, il est alimenté par quatre piles de format AA qui sont placées dans ce logement. Reportez-vous au chapitre 2 - Alimentation du PC-1600.

3 Logements des modules Les modules d'extension de RAM en option peuvent être enfichés dans ces deux logements pour augmenter la capacité de la mémoire, des programmes de sauvegarde et de chargement ou créer une disquette de RAM.

Le clavier du PC-1600

Cette section présente certaines des caractéristiques importantes du clavier du PC-1600. Les fonctions des touches se rapportant au BASIC seront décrites ultérieurement dans ce manuel, au chapitre 9, les modes d'exploitation en BASIC, dans la section Fonctionnement des touches et l'utilisation de la touche **CTRL** (contrôle) dans la section Mode d'édition. La visualisation d'une simple image-écran est également décrite dans le chapitre 4 sur l'Affichage du PC-1600.



Le clavier du PC-1600

1Touches alphabétiques Les touches alphabétiques sont identiques à celles d'une machine à écrire habituelle et sont utilisées pour introduire les programmes et les données. Ces touches ont une fonction d'autorépétition qui répétera continuellement le caractère de la touche que vous maintiendrez enfoncée. Une commande BASIC vous permet de mettre en ou hors circuit cette fonction. Se reporter à KEYSTAT dans le Dictionnaire des commandes.

2Touche des minuscules (Bas de casse) La touche **SML** (minuscules) vous permet de commuter la valeur par défaut des majuscules en minuscules lorsque vous tapez sur les touches alphabétiques. Cette touche inverse également la

fonction de la touche SHIFT comme décrit ci-dessus. Après avoir pressé la touche SML et que vous êtes en mode des minuscules, il faut presser la touche avant SHIFT chaque entrée de caractères pour produire un caractère en majuscule.

3 Touches des majuscules/de décalage (Haut de casse) Pour une grande praticité, le PC-1600 est équipé de deux touches SHIFT; elles ont toutes deux des fonctions identiques. Comme sur une machine à écrire, une pression sur la touche avant de presser une autre touche donne un autre caractère, une autre fonction ou un SHIFT autre symbole. Normalement, le PC-1600 imprime les caractères en majuscules. Il s'agit d'une valeur par défaut de l'ordinateur, à moins que vous ne le spécifiiez autrement. Vous pouvez obtenir les minuscules en pressant la touche avant d'appuyer sur chaque touche de caractère. (Reportez-vous à la description de la touche SML (minuscules) qui précède. Les touches sont également d'autres fonctions spéciales quand elles sont utilisées en combinaison avec d'autres touches dans le BASIC. Ces fonctions sont décrites dans la section Référence du langage BASIC.

4 Touches de fonctions Ces six touches sont en haut du clavier, juste en-dessous de l'affichage et sont utilisées pour assigner ou rappeler avec quelques mots-clés ou commandes utilisés fréquemment dans les manipulations. L'utilisation de ces touches et leurs combinaisons avec d'autres touches s'y rapportant sont décrites dans la section des Fonctions des touches en BASIC.

5Touches de mise sous/hors tension Ces touches sont utilisées pour mettre l'ordinateur sous et hors tension. Quand l'ordinateur fonctionne, la touche ON est utilisée en tant que touche BREAK qui a pour but d'interrompre l'exécution d'un programme.

6Touche d'effacement Cette touche efface la ligne sur lequel le curseur est placé et dispose également d'autres caractéristiques quand elle est changée en touche CA en BASIC. Se reporter à la description des touches dans la section Référence du langage Basic.

7 Touche de mode Cette touche est utilisée pour passer du mode d'exploitation PROGRAM au mode d'exploitation RUN du PC-1600. Le troisième mode, RESERVE, est sélectionné quand la touche MODE est spécifiée(SHIFT + MODE).

8 Touches des opérations arithmétiques et numériques Ces touches sont utilisées pour exécuter des calculs directs et pour introduire des données numériques dans les programmes. Les signes +, −, *, / sont équivalents à l'addition, la multiplication et la division. L'utilisation de ces touches est expliquée d'une manière plus détaillée dans le chapitre 5, Calculs sur le PC-1600.

9 Touche d'introduction. La pression de la touche ENTER après d'autres manipulations signale à l'ordinateur la fin de l'entrée. La fonction de cette touche peut être comparée à la touche Retour utilisée sur les machines à écrire.

Vue d'ensemble du logiciel

Votre ordinateur de poche PC-1600 est basé sur une version proche du Z-80A. Ce microprocesseur fonctionne à partir de commandes écrites en code machine. Les instructions de programmation BASIC que vous écrivez et introduisez dans le PC-1600 sont traduites internement ligne par ligne tandis que votre programme est exécuté. Ce procédé est effectué par l'interpréteur BASIC incorporé dans la mémoire morte (ROM) de l'ordinateur. Sauf en ce qui concerne quelques commandes et fonctions de calculs internes qui travaillent directement dans les sous-programmes du code machine, le BASIC est le langage avec lequel vous communiquerez vos instructions au PC-1600.

La version du langage BASIC utilisée dans le PC-1600 est le développement par SHARP d'un BASIC spécialement perfectionné pour les ordinateurs de poche.

Les programmes écrits en BASIC pour les autres ordinateurs peuvent habituellement, par une adaptation avec de petites modifications et un peu d'ingéniosité, être exécutés sur le PC-1600. Il y a une section séparée dans ce manuel sur la compatibilité avec la version du BASIC utilisée sur le PC-1500, ainsi que la liste des commandes qui doivent être modifiées ou changées.

Votre PC-1600 ne fera pas office de traitement de texte dans sa présentation de la feuille, mais les descriptions des instructions LOAD* et SAVE* comprennent des recommandations sur la manière d'utiliser les lignes de programme BASIC pour mémoriser un texte plutôt que les programmes. Dans ce sens, le PC-1600 constitue une base de données compacte et puissante. Naturellement, un programme éditeur de texte en BASIC vous permettra d'éditer un texte aussi bien que de le mémoriser. Il y a de nombreux livres excellents de disponibles dans le commerce avec des idées sur des projets de programmation utiles en addition aux programmes d'exemples compris dans ce manuel.

Pour les programmeurs chevronnés, il est possible d'annuler la fonction de l'interpréteur BASIC et d'écrire les programmes directement dans le langage machine. Il y a un certain nombre de commandes dans cet ensemble qui vous permettent de manipuler les programmes en langage machine (PEEK, POKE, CALL, BLOAD, BSAVE, etc.). Ces commandes doivent être utilisées avec précautions et seulement par ceux qui ont une compréhension totale de l'allocation mémoire du PC-1600.

PARTIE II

FONCTIONNEMENT

Le premier chapitre de cette partie commence par les aspects pratiques d'utilisation de votre PC-1600.
Vous apprendrez comment alimenter votre ordinateur, ce qu'il se passe lorsque vous le mettez sous ou hors tension, comment le remettre à zéro pour initialiser le système et en cas de problème.

Le chapitre suivant traite des caractéristiques de l'affichage du PC-1600 et vous montre comment régler la date et l'heure de l'horloge en temps réel incorporée.

Le dernier chapitre décrit comment utiliser le PC-1600 en tant que calculatrice et exécuter de simples opérations.

2 Alimentation du PC-1600

Votre PC-1600 peut fonctionner sur piles comme sur secteur avec un adaptateur CA en option.

Mise en place des piles

Quatre piles de 1,5 V, taille AA sont comprises en tant qu'accessoire standard avec le PC-1600. Ce sont des piles sèches ordinaires qui ne sont pas rechargeables.

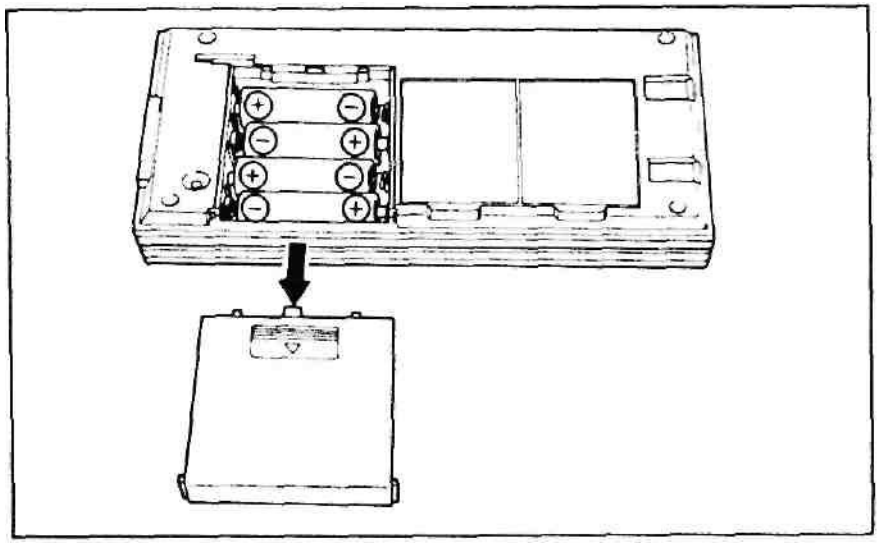
La longévité des piles dépend de la fréquence avec laquelle vous utilisez votre ordinateur et dans quelles conditions.

Habituellement, en faisant normalement 50 mn d'affichage et 10 mn de programmation par heure, les piles durent environ 25 heures à 20°C.

Pour mettre en place les piles

1Enlevez le couvercle du logement des piles en le faisant glisser dans le sens de la flèche.

2Insérez les piles en respectant la polarité +/- comme indiqué sur l'illustration.



Dépose du couvercle et mise en place des piles

3Remettez en place le couvercle du logement des piles.

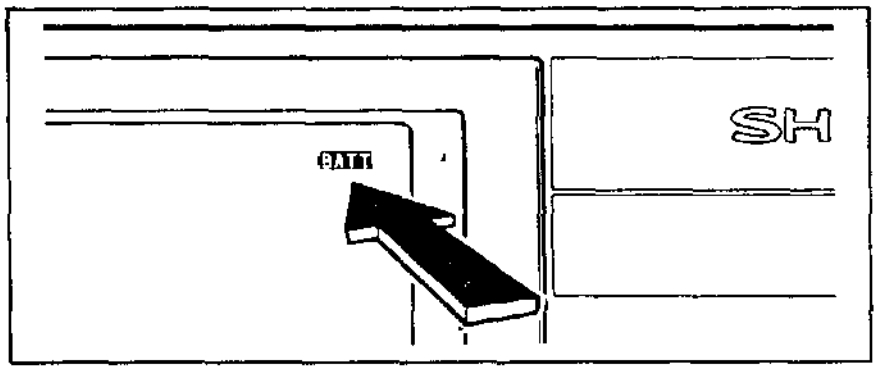
4Reportez-vous à la section Réinitialisation de l'ordinateur du chapitre 3 et suivez la procédure de ALL RESET pour initialiser l'ordinateur.

Pour toute information sur le remplacement des piles, voyez l'Annexe A.

Témoin d'épuisement des piles

Un témoin d'épuisement des piles apparaît sous le symbole BATT dans la ligne d'état en haut de l'affichage du PC-1600 dès que la puissance des quatre piles devient faible.

Le symbole BATT peut également être affiché quand l'ordinateur est connecté a un périphérique alimenté par piles, tel que l'imprimante CE-1600, dont les piles sont épuisées.



Si l'imprimante est connectée au PC-1600 quand le symbole BATT apparaît, vous pouvez savoir si les piles déficientes sont celles de l'ordinateur ou de l'imprimante, en pressant SHIFT + ON.

- Si le symbole BATT est encore affiché après cette opération, les piles de l'ordinateur sont déficientes.

Vous devez alors remplacer les piles du PC-1600 le plus tôt possible. Reportez-vous à l'Annexe A, Remplacement des piles. Attention : le remplacement des piles implique un risque de perte des programmes et des données. Lisez attentivement cette section.

Vous pouvez également, à ce moment, choisir d'utiliser un adaptateur CA pour alimenter l'ordinateur si le remplacement des piles n'est pas possible. L'utilisation de l'adaptateur est décrite dans la section suivante, comme dans les pages sur le remplacement des piles. Si vous essayez d'utiliser l'ordinateur tandis que le témoin BATT est affiché, le PC-1600 sera automatiquement commuté hors tension.

o Si le symbole BATT n'est plus affiché après que vous avez pressé SHIFT + ON,
Les piles déficientes sont celles de l'imprimante (ou tout autre périphérique connecté).

En plus du symbole BATT, votre PC-1600 dispose également de deux autres sortes d'avertissements d'épuisement des piles quand les équipements en option sont connectés à l'ordinateur :

- Quand l'ordinateur est mis sous tension, un message, CHECK _____, Apparaît sur l'écran suivi du code numérique de l'équipement suspect.

- Durant l'exécution d'un programme, un code ERROR (d'erreur) est émis, là encore avec un chiffre identifiant la cause du problème.

Les messages CHECK sont traités dans la section de la Mise sous tension au Chapitre 3 et les codes ERROR pour les équipements E/S individuels, sont décrits dans la PARTIE III, Utilisation des interfaces et des équipements E/S en option. Les codes ERROR sont aussi résumés dans un tableau de référence de l'Annexe F.

Si l'imprimante est en cause, ses piles au nickel-cadmium incorporées doivent être rechargées. Référez-vous aux instructions du manuel de l'imprimante pour recharger les piles.

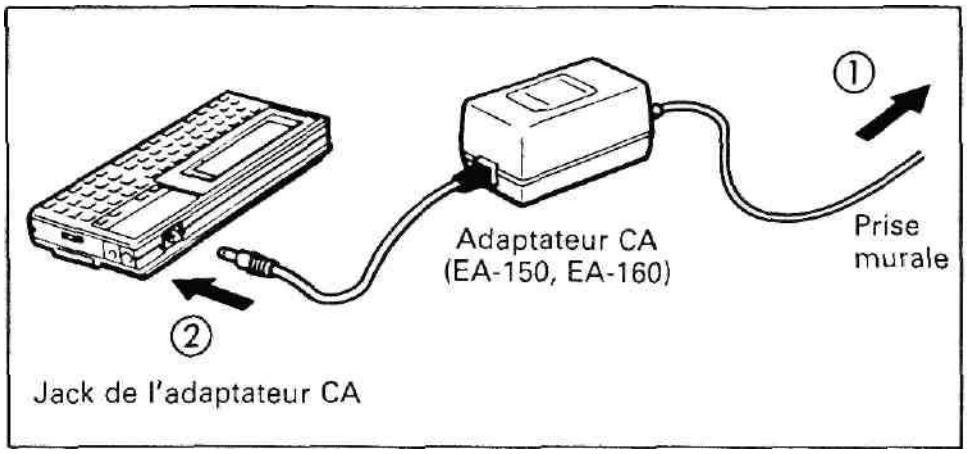
Vous pouvez voir par une double vérification que les piles déficientes sont dans l'imprimante et non pas dans l'ordinateur lui-même, en essayant d'obtenir un message CHECK sur l'affichage en mettant l'ordinateur hors tension, puis sous tension à nouveau. Quand le message CHECK apparaît, reportez-vous aux pages mentionnées ci-dessus.

Branchement sur le secteur

En utilisant un adaptateur CA connecté sur le jack au dos du PC-1600, l'ordinateur peut être alimenté en intérieur à partir d'une prise électrique murale normalisée ou pour économiser les piles.

Deux adaptateurs CA sont disponibles en tant qu'option individuelle ou en tant qu'accessoires standard avec les imprimantes suivantes:

- EA-160 (avec l'imprimante CE-1600P)
- EA-150 (avec l'imprimante CE-150)



Connexion du PC-1600 au secteur par adaptateur

Pour raccorder le PC-1600 au secteur

1Mettez le PC-1600 et les équipements périphériques qui lui sont raccordés hors tension.

2Tout d'abord branchez le cordon de l'adaptateur CA dans la prise électrique murale.

3Puis raccordez le cordon de l'adaptateur au jack situé au dos du PC-1600.

4Assurez-vous de bien procéder en sens inverse en déconnectant l'adaptateur de l'ordinateur. Ne débranchez jamais l'adaptateur directement du jack de l'ordinateur lorsque l'ordinateur est encore sous tension.

5Débranchez l'adaptateur de la prise électrique murale en cas de non utilisation.

IMPORTANT: Si les piles de l'ordinateur sont épuisées ou absentes quand l'adaptateur CA est déconnecté de l'ordinateur, toutes les données restant dans la mémoire de l'ordinateur sont perdues.

L'adaptateur CA est également utilisé pour recharger les piles NiCad incorporées dans l'imprimante et les autres équipements et l'utilisation d'un adaptateur CA est un moyen de préserver les données dans la mémoire de l'ordinateur quand vous remplacez les piles. Reportez-vous au Remplacement des piles, de l'Annexe A.

3 Mise sous et hors tension

Ce chapitre décrit les méthodes pour mettre votre PC-1600 sous et hors tension. Vous verrez également des exemples de l'affichage LCD après la mise sous tension, dans diverses conditions de fonctionnement.

La section suivante mentionne l'utilisation de modules d'extension de RAM et autres équipements périphériques E/S avec l'ordinateur PC-1600.

Ces équipements et modules de mémoire sont couverts dans la Partie III, Utilisation des interfaces et des équipements E/S en option. Une section sur le Remplacement des modules de RAM est incluse dans l'Annexe B. De plus, référez-vous toujours aux manuels, accompagnant toutes les options disponibles pour cet ordinateur, pour des informations détaillées spécifiques.

Mise sous tension

Vous pouvez mettre votre PC-1600 sous tension de trois manières différentes:

- Manuellement, en pressant la touche **ON**.

- En utilisant la commande WAKE$ en BASIC pour commuter l'ordinateur à une date et une heure spécifiées. (Reportez-vous à WAKE$ dans le Dictionnaire des commandes.)

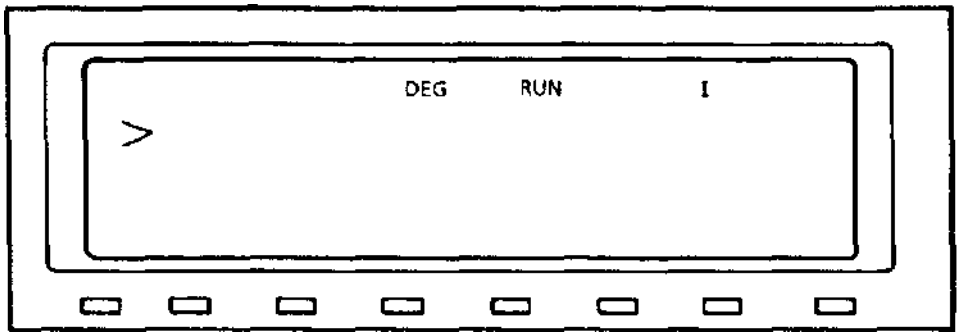
- A nouveau à l'aide de la commande WAKES, en programmant le PC-1600 pour se commuter lui-même quand un signal CI est reçu, via le port série RS-232C de l'ordinateur en provenance d'un équipement extérieur tel qu'un téléphone avec modem. (Reportez-vous à la commande WAKES et aux spécifications d'interface en série.)

Ce que vous voyez sur l'écran, lorsque vous mettez votre PC-1600 sous tension, dépend si votre ordinateur est utilisé individuellement ou avec des modules de RAM en option montés dans les logements d'extension, ou avec d'autres équipements E/S connectés. Votre PC-1600 est doté de messages et de symboles d'avertissement pour vous alerter de l'épuisement des piles et des autres problèmes à la mise sous tension.

Les exemples suivants montrent l'écran LCD, après la mise sous tension, dans ces différentes conditions.

• Utilisation du PC-1600 sans options, dans une condition de mise sous tension normale...

Quand vous pressez la touche **ON** l'écran affiche ceci :



Tout en haut de l'affichage se trouve la ligne d'état qui montre les fonctions et le mode d'exploitation que vous avez sélectionnés. Quand vous mettez votre PC-1600 sous tension pour la toute première fois ou après une réinitialisation, la ligne d'état affiche DEG (degrés), RUN (mode d'exploitation RUN) et I (Mode I de RESERVE).

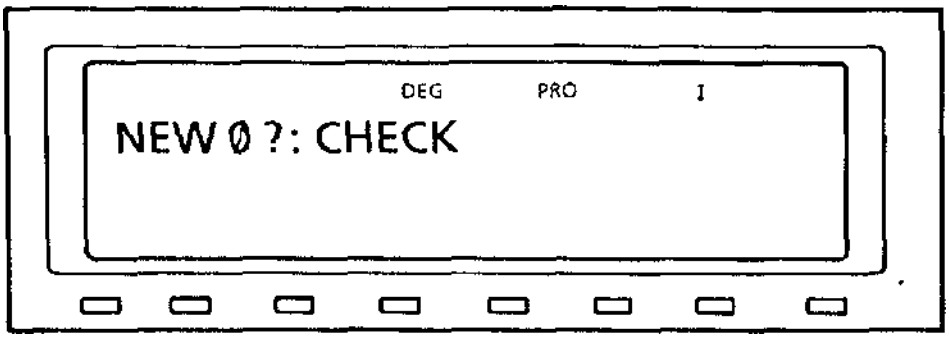
Ces caractéristiques sont également décrites dans les sections BASIC et Affichage du PC-1600.

La ligne d'état apparaît chaque fois que l'ordinateur est mis sous tension et reste affichée pendant le fonctionnement. Les symboles de cette ligne montrent les valeurs en vigueur la dernière fois que l'ordinateur a été mis hors tension, par conséquent elles peuvent être différentes chaque fois que vous mettez votre PC-1600 sous tension.

Le symbole > est appelé caractère de sollicitation du système et indique que l'ordinateur attend que vous tapiez sur cette ligne. Le caractère de sollicitation est remplacé par le premier caractère que vous introduisez sur la ligne.

Après installation ou remplacement d'un module de RAM...

Vous devez mettre l'ordinateur hors tension avant d'installer ou de remplacer les modules de RAM en option. Quand vous mettez à nouveau l'ordinateur sous tension, l'écran affiche :



Ce message vous indique que l'ordinateur reconnaît la mémoire additionnelle et attend que vous effaciez dans la zone mémoire toutes les fausses données restantes. Il est à noter que ce message n'apparaît que lorsque les modules de mémoire sont montés comme mémoire d'extension. Le message n'est pas affiché si un module de programme s'articule en tant que disque de RAM ou en mémoire de programme. L'utilisation de ces modules est décrite dans la PARTIE III de ce manuel.

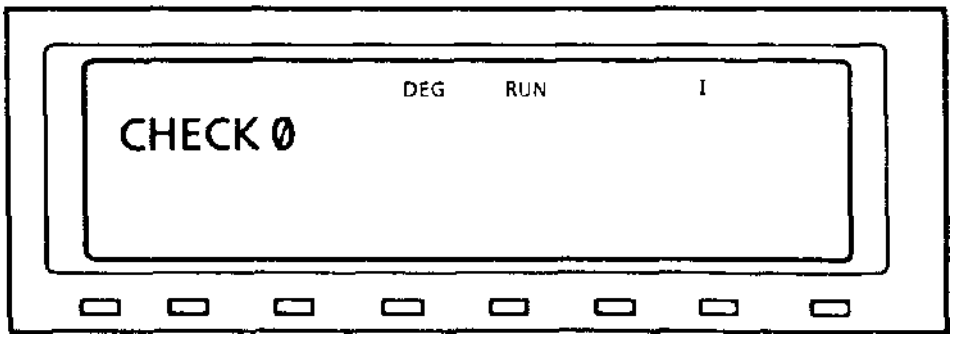
Pour vider la mémoire de l'ordinateur et restaurer les valeurs par défaut, ou pré spécifiées, pressez la touche **CL** (remise à zéro). Assurez-vous que le mode d'exploitation est positionné sur PROgramme. Pressez la touche **MODE** si nécessaire. Puis tapez N E W 0 et pressez **ENTER**.

L'affichage doit alors revenir sur le caractère de sollicitation > indiqué sur l'illustration précédente et vous pouvez alors commencer à utiliser l'ordinateur.

Quand les modules de RAM ne sont pas installés correctement...

Il y a deux logements d'extension, S1 et S2, au-dessous du PC-1600, dans lesquels vous pouvez connecter les modules de RAM en option pour l'adjonction supplémentaire de mémoire dans l'ordinateur ou de stockage des programmes et données. Ces modules sont décrits en détails dans la PARTIE III, Utilisation des interfaces et des équipements E/S en option, et dans l'Annexe B pour leur remplacement.

Trois de ces modules en option, le CE-151, le CE-155 et le CE-159 ne peuvent être utilisés que dans le logement 1. Comme les deux logements sont identiques dans leur dimension comme dans leur forme, si vous connectez par inadvertance l'un de ces modules dans le logement 2, l'ordinateur ne pourra pas l'identifier. Lors de la mise sous tension, l'affichage indiquera :



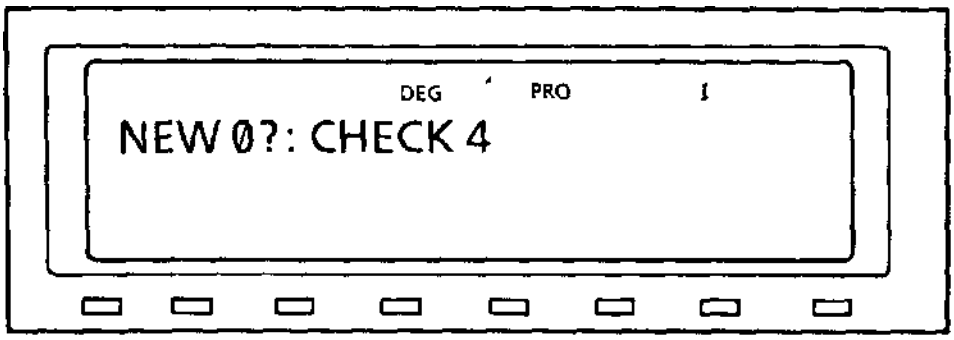
Ce message CHECK est une caractéristique particulière au PC-1600 pour vous alerter que quelque chose ne va pas et vous dire de vérifier la cause suspecte du problème.

Le nombre suivant le mot CHECK est un code signalant les problèmes des équipements. Le 0 dans ce cas signifie le logement 2. Un tableau de ces codes CHECK est donné dans la section suivante.

Vous devez éliminer le problème signalé par le message CHECK avant de continuer à utiliser l'ordinateur.

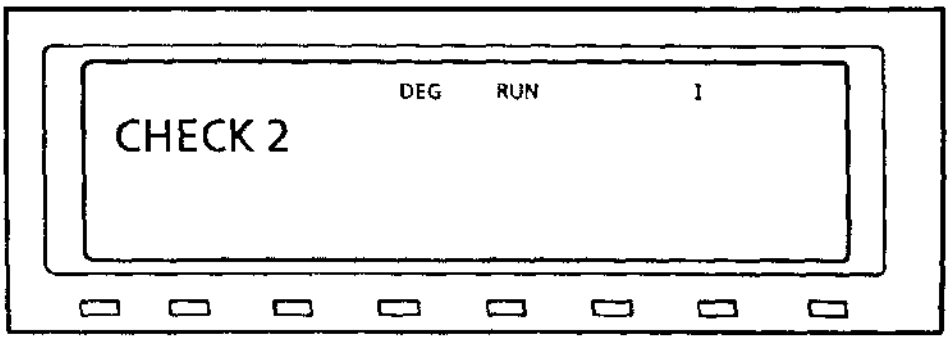
Quand des équipements E/S avec des piles déficientes sont connectés...

Certains des équipements périphériques en option qui sont connectés sur votre PC-1600 ont leurs propres piles incorporées. Si les piles sont déficientes dans l'équipement connecté à l'ordinateur, les messages suivants apparaîtront lorsque vous mettrez votre PC-1600 sous tension.



N E W 0 ? vous avertit que l'ordinateur reconnaît un changement dans la mémoire et est un caractère de sollicitation pour vider la mémoire.

Ou, en tant qu'autre possibilité.



Ces messages CHECK vous disent que les piles des équipements connectés ont besoin d'être rechargées ou remplacées. De plus, comme décrit dans la section précédente, CHECK 0 est un avertissement que le mauvais module mémoire est connecté dans le logement 2.

Reportez-vous au tableau suivant pour la signification de ces messages CHECK illustrés ci-dessus et des autres codes CHECK qui peuvent apparaître. Les numéros de code manquant dans la liste sont réservés pour les équipements ultérieurs, qui peuvent être disponibles dans l'avenir.

Code CHECK	Equipement connecté	Cause
0	Logement de module 2	CE-151, CE-155 ou CE-159 incorrectement montés. Utiliser seulement le logement 1 pour ces modules.
2	Unité de disque CE-1600F	Se connecte sur l'imprimante CE-1600P. Les piles niCad incorporées sont déficientes ou le matériel de l'imprimante est défectueux.
4	Imprimante CE-1600P	Les piles NiCad incorporées sont déficientes ou le matériel de l'imprimante est défectueux.
6	Imprimante CE-150	Les piles NiCad incorporées sont déficientes ou le matériel de l'imprimante est défectueux.
8	Interface parallèle/ séquentielle CE-158	Les piles NiCad incorporées sont déficientes.

Mise hors tension

Votre ordinateur PC-1600 dispose de deux caractéristiques de mise hors tension automatique, pour s'éteindre de lui-même. La caractéristique Sauvegarde de mémoire du PC-1600 préserve toutes les données sur lesquelles vous avez travaillé quand l'ordinateur est mis hors tension, que ce soit manuellement ou automatiquement.

Le PC-1600 se met automatiquement hors tension dans les conditions suivantes :

Pour économiser les piles, le PC-1600 se met lui-même hors circuit quand il n'y a eu aucune introduction de touche pendant dix minutes. Ceci ne se produira pas si vous exécutez un programme durant cet intervalle de temps. Pressez simplement la touche **ON** pour recommencer. Cette fonction peut être invalidée par l'instruction POWER du BASIC.

Si vous essayez de continuer à faire fonctionner l'ordinateur après que le témoin de piles déficientes BATT apparaît, le PC-1600 s'éteint de lui-même pour éviter des erreurs de traitement et une perte éventuelle des données. Quand BATT est affiché, vous devez mettre l'ordinateur hors tension et remplacer les piles ou utiliser l'adaptateur CA en option aussitôt que possible.

Dans des conditions normales de fonctionnement, vous disposez de deux manières pour mettre votre ordinateur PC-1600 hors tension:

- Manuellement, en pressant la touche **OFF**.

- En utilisant la commande POWER du BASIC. Reportez-vous au Dictionnaire des commandes en BASIC pour les détails.

Réinitialisation de l'ordinateur

Il arrive des moments où il est nécessaire de vider la mémoire de l'ordinateur, de réinitialiser ses valeurs par défaut ou préspécifiées et de remettre l'ordinateur en condition exploitable.

Cette action est appelée réinitialiser l'ordinateur et il y a deux niveaux de réinitialisation pour votre PC-1600: simple et ALL RESET (réinitialisation générale).

La réinitialisation simple est réalisée en pressant le bouton RESET sous l'ordinateur. La commande ALL RESET est exécutée en pressant et maintenant la touche **ON** (BREAK) tout en enfonçant le bouton RESET.

La réinitialisation est principalement nécessaire dans deux conditions:

- Pour initialiser l'ordinateur et le préparer à l'exploitation après la mise en place ou le remplacement des piles.

- Pour restaurer l'ordinateur dans une condition d'exploitation normale après qu'il se soit "bloqué".

Si l'ordinateur est soumis à un fort parasitage électrique ou s'il fonctionne dans les conditions instables qui peuvent conduire à des erreurs de traitement, il peut se "bloquer" et cesser de fonctionner normalement.

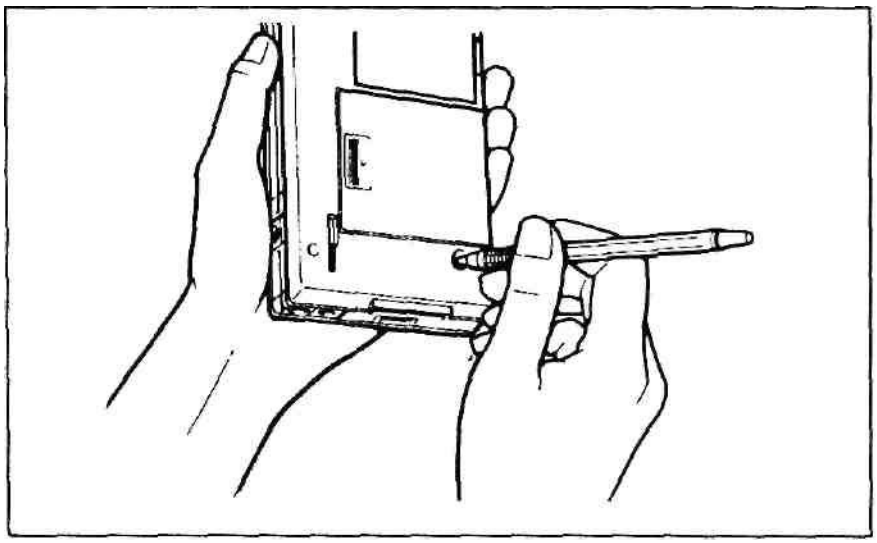
Les effets peuvent varier, mais l'écran sera "gelé" ou sera effacé, l'opération des touches peut provoquer des caractères bizarres et incohérents, ou les touches peuvent être complètement inopérables. La simple mise hors tension puis sous tension de l'ordinateur n'est pas possible si les touches **OFF** et **ON** ne fonctionnent pas.

Un bouton RESET figure à l'arrière de l'ordinateur pour libérer cette condition. Ce bouton est profondément encastré pour éviter son utilisation accidentelle car la fonction de réinitialisation est très puissante et doit être utilisée avec précaution.

ATTENTION: Les programmes et données peuvent être détruits!

Utilisez un objet pointu comme la pointe d un stylo a bille pour enfoncer le bouton RESET. N'utilisez pas de pointes qui se cassent facilement comme des mines de crayons noirs ou des épingles.

Si votre ordinateur est connecté à l'imprimante CE-1600P en option, vous n'avez pas besoin de débrancher l'ordinateur pour accéder au bouton RESET situé derrière. Un bouton RESET identique figure également à l'arrière du berceau de l'imprimante et peut être utilisé à la place, pour la procédure qui suit.



Pression sur le bouton RESET

Les instructions suivantes donnent les phases à suivre et montre l'état de l'affichage quand vous réinitialisez votre ordinateur PC-1600.

Si l'ordinateur se "bloque", essayer tout d'abord la réinitialisation simple, car son effet est identique à la mise hors et sous tension et, dans la plupart des cas, les programmes et données seront récupérés en toute sécurité, sinon dans leur totalité. La procédure ALL RESET est générale et ne doit être utilisée qu'en dernier ressort, quand toutes les autres mesures ont échoué.

ALL RESET est également utilisé pour initialiser l'ordinateur après avoir installé et remplacé les piles.

Le tableau suivant donne la comparaison des différents effets des actions de réinitialisation simple et ALL RESET sur les valeurs par défaut et les paramètres du système de l'ordinateur.

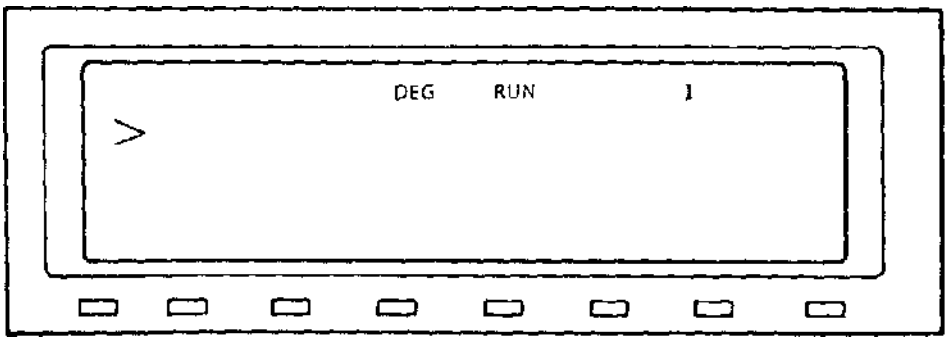
Résumé des paramètres de réinitialisation

	ALL RESET	Réinitialisation simple ou mise sous tension
Mise sous tension auto. (WAKES)	OFF	Valeur préservée
Mise hors tension auto.	10 mn	Valeur préservée
TITLE	S0:	Valeur préservée
Date/heure	KM) 1(J)0(H)0(mr i) 0(sec)	Valeur préservée
Touches de fonction	Effacées	Valeur préservée
Mot de passe	Annulé	Valeur préservée
Accès séquentiel RS-232C	COM1 1200,8,N,1,X,S COM2 38400,7,E,2,X,S	Valeur préservée
Zone langage machine	0	Valeur préservée
Cliquetis de touche	OFF	Valeur préservée
BEEP	ON	Valeur préservée
Répétition de touche	OFF	Valeur préservée
Instructions d'interruption	Invalide	Valeur préservée
Mode	RUN DEG I	Préservé ou changé après que l'erreur est générée
BREAK ON/OFF	ON	Valeur préservée
LOCK/UNLOCK	UNLOCK	Valeur préservée
Mode 0/1	Mode 0	Valeur préservée
MAXFILES	0	Valeur préservée
Module extension de mémoire	Mémoire effacée	Mémoire préservée
Module programme	Mémoire préservée	Mémoire préservée
Module disque de RAM	Mémoire préservée	Mémoire préservée
RAM interne	Mémoire effacée	Mémoire préservée

Pour exécuter une réinitialisation simple

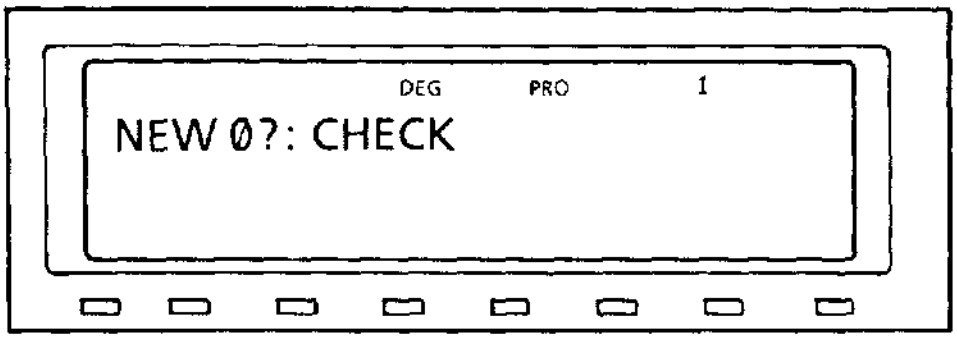
1 Repérez le bouton RESET sur le dos de l'ordinateur et enfoncez-le pendant deux secondes.

2 L'écran doit être nettoyé et afficher le caractère de solicîtation du système >, comme illustré ci-dessous :



3Le caractère de sollicitation signifie que l'ordinateur est prêt à recevoir votre introduction. Si l'ordinateur n'est pas vidé et s'il n'y a aucun changement sur l'écran, enfoncez à nouveau le bouton RESET.

4Après avoir pressé le bouton RESET, l'affichage est quelques fois le suivant lorsque l'ordinateur est utilisé avec les équipements E/S en option :



5 Pour vider l'ordinateur de cet écran, pressez la touche **CL** (nettoyage). Puis assurez-vous que le mode d'exploitation est spécifié sur PROgramme. Pressez la touche **MODE** si nécessaire. Puis tapez N E W 0 et pressez la touche **ENTER**.

L'affichage doit alors revenir sur le caractère de sollicitation du système > indiqué sur l'illustration précédente et vous pouvez commencer à utiliser l'ordinateur à nouveau.

Pour exécuter une réinitialisation générale (ALL RESET)

Après avoir mis en place ou remplacé les piles, procédez comme suit.

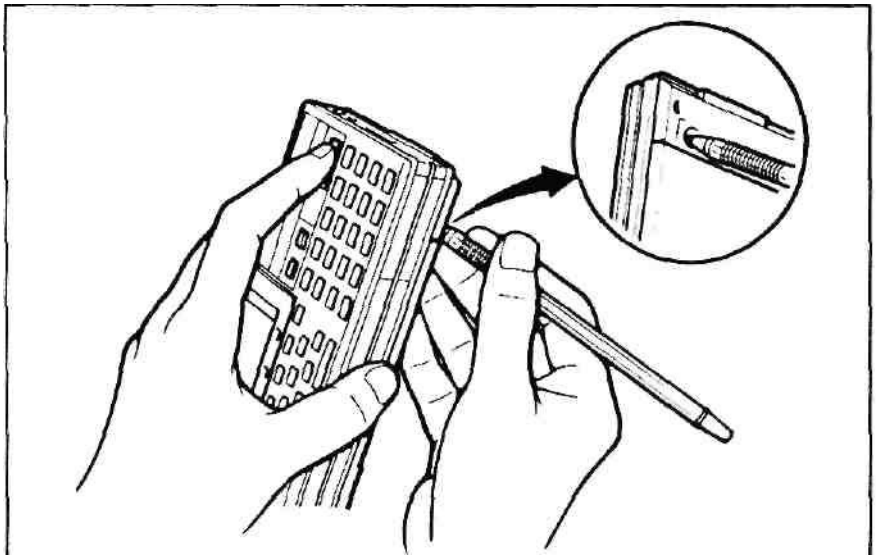
Dans tous les autres cas, lisez les paragraphes suivants avant de continuer :

IMPORTANT: Si l'ordinateur s'est bloqué durant son fonctionnement, essayez la réinitialisation simple avant d'aborder les phases suivantes. ALL RESET est une action complète qui peut détruire les données et les programmes. Vérifiez le tableau des paramètres de réinitialisation dans cette section et assurez-vous que vous pouvez prendre ce risque. Les données et programmes dans la mémoire d'extension et dans la mémoire interne RAM sont complètement remis à zéro après ALL RESET

1Tenez l'ordinateur de façon à avoir un accès facile à la touche **ON** à droite du clavier. Notez que lorsque l'ordinateur est sous tension, la touche **ON** fonctionne en tant que touche BREAK.

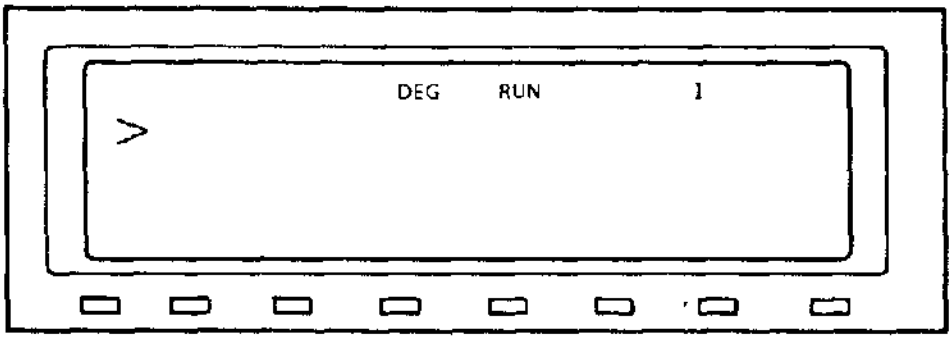
2Pressez et maintenez la touche **ON** (BREAK), puis enfoncez le bouton RESET à l'arrière de l'ordinateur.

3 Relâchez le bouton RESET tout d'abord, puis relâchez la touche EEJ (BREAK), comme illustré ci-dessous.



ALL RESET à l'aide du bouton RESET et de la touche ON (BREAK)

4 L'écran doit afficher le caractère de sollicitation > comme illustré ci-dessous:



5 Le caractère de sollicitation signifie que l'ordinateur est prêt à l'utilisation et attend que vous procédiez à une entrée. Si le caractère de sollicitation n'est pas affiché, enfoncez le bouton RESET et la touche ON (BREAK) à nouveau, comme
décrit aux phases de 1 à 3.

IMPORTANT: Si vous ne pouvez pas réinitialiser l'ordinateur après avoir essayé soit la réinitialisation simple ou ALL RESET ou ies deux, contactez votre concessionnaire SHARP immédiatement.

4 L'affichage du PC-1600

L'affichage sur votre ordinateur PC-1600 est un affichage à cristaux liquides (LCD) de 26 caractères x 4 lignes, avec contraste réglable pour une lecture optimale.

Deux modes écran peuvent être spécifiés par commande BASIC. En MODE 0, l'écran peut être complètement utilisé sur 4 lignes, affichant 26 caractères par ligne. Le MODE 1 est une émulation de l'affichage à 1 ligne de l'Ordinateur de poche PC-1500 lors de l'utilisation des instructions PRINT et GPRINT pour la compatibilité avec ses programmes et données. Quand la longueur de l'introduction excède la longueur physique de la ligne en ce mode, l'écran roule vers le haut à partir de la dernière ligne.

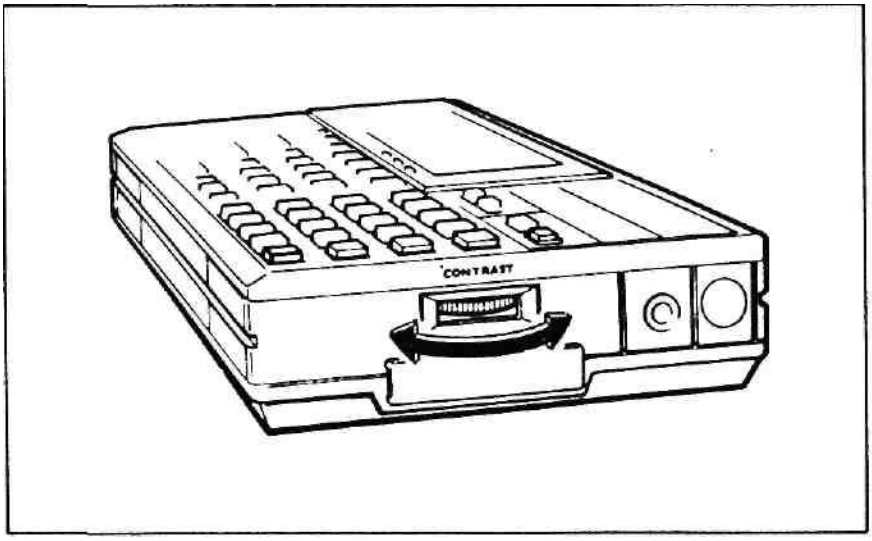
Le MODE 0 permet à l'affichage d'être utilisé en mode graphique dans lequel un écran total de 156 x 32 points est disponible pour l'affichage des données graphiques d'images binaires.

Ces modes sont décrits en détail dans le chapitre Référence du language BASIC de ce manuel.

Réglage du contraste d'affichage

Lorsque les conditions de luminosité changent ou que vous changez l'orientation de l'ordinateur en lui connectant l'imprimante en option, vous pouvez avoir besoin de régler le contraste de l'affichage pour une meilleure vision.

La molette de réglage de contraste est située à gauche de l'ordinateur, just en dessous de l'accès séquentiel RS-232C.



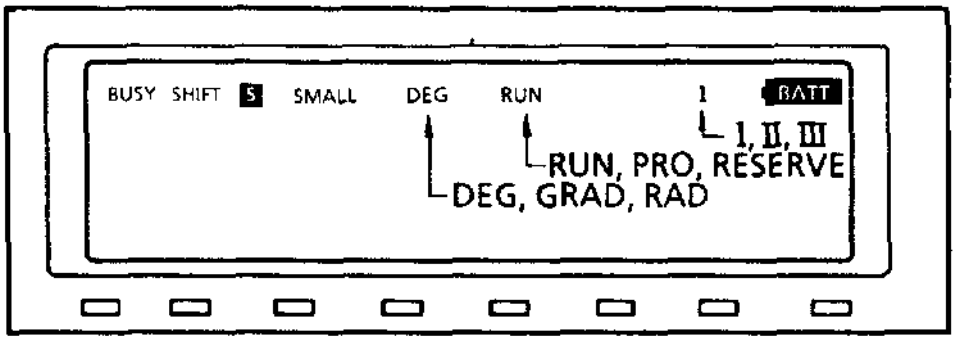
Molette de réglage du contraste de l affichage

Tournez la molette comme désiré pour éclairer ou assombrir l'écran jusqu'à ce que vous puissiez lire facilement. Réglez la molette à nouveau quand vous déplacez l'ordinateur ou que vous changez son orientation.

Symboles de la ligne d'état

Une ligne d'état spéciale figure tout en haut de l'écran du PC-1600, au-dessus de la première des quatre lignes d'affichage, qui affiche les symboles pour l'épuisement des piles, le mode d'exploitation et les fonctions que vous avez sélectionnées ou qui sont actuellement utilisées.

La ligne d'état vous permet une vérification rapide, d'un simple coup d'œil, des fonctions que vous avez sélectionnées si vous les avez spécifiées correctement et de ce que fait l'ordinateur en ce moment.



Symboles de la ligne d'état de l'affichage

Les symboles dans la ligne d'état de l'affichage ne changent pas, parce que ce sont des valeurs qui restent effectives lorsque vous remettez votre ordinateur sous tension après l'avoir mis hors tension. Ceci vous permet de continuer avec les valeurs précédentes telles quelles, ou après avoir vérifié la ligne d'état, de faire le changements nécessaires pour continuer à partir de ce moment.

Voici une liste des divers symboles qui sont affichés dans la ligne d'état et leur signification.

BUSY	Affiché tandis que l'ordinateur exécute un programme ou une commande. Une autre introduction de touche n'est pas possible à ce moment.
SHIFT	Affiché à chaque pression de la touche **SHIFT**. S'efface lors de l'introduction de la touche suivante.
DEG	Indique que le mode Degré est sélectionné pour une fonction angulaire.
RAD	Indique que le mode Radian est sélectionné pour une fonction angulaire.

GRAD	Indique que le mode Gradient est sélectionné pour une fonction angulaire.
RUN	Affiché lorsque l'ordinateur est en mode RUN pour effectuer des calculs directs ou exécuter un programme.
SMALL	Affiché quand l'ordinateur est en mode des minuscules pour taper les caractères en minuscules sur l'écran.
PRO	Affiché quand l'ordinateur est en mode PROgramme pour la création, l'édition ou le listage des programmes.
RESERVE	Affiché quand l'ordinateur est en mode RESERVE pour assigner et éditer des chaînes de caractères aux touches de fonction.
DEF	Affiché lors de chaque pression de la touche DEF (de définition). S'efface lors de l'introduction de la touche suivante.
I	Indique la sélection de la chaîne de caractères dans le menu I du mode RESERVE.
II	Indique la sélection de la chaîne de caractères dans le menu II du mode RESERVE.
III	Indique la sélection de la chaîne de caractères dans le menu II du mode RESERVE.
CTRL	Affiché lors de chaque pression de la touche CTRL (touche de service). S'efface lors de l'introduction de la touche suivante.
BATT	Indique l'épuisement des piles dans l'ordinateur PC-1600 ou dans le périphérique connecté.
S	Indique la sélection du mode II du clavier (KB II) pour l'utilisation de la grille du jeu de caractères internationale.

Mode d'exploitation

Votre PC-1600 dispose de trois modes d'exploitation: RUN, PROGRAMME et RESERVE. Pour sélectionner le mode RUN ou le mode PROGRAMME, pressez la touche MODE pour basculer, ou commuter par intermittence, d'un mode à l'autre.
Le mode RESERVE est sélectionné en pressant les touches SHIFT + MODE.

Le symbole de chaque mode que vous sélectionnez est affiché dans la ligne d'état en haut de l'écran.

Vous pouvez aussi spécifier le mode d'exploitation avec deux commandes BASIC, LOCK et PASS, lors de la programmation sur votre ordinateur. Reportez-vous le Dictionnaire des commandes pour voir comment elles se rapportent à l'utilisation des touches.

L'utilisation de ces modes d'exploitation est décrit dans la section de Référence du langage BASIC. Voici une brève description de chacun des modes et quand il est utilisé:

RUN Le mode RUN est pour les calculs manuels, ou directs, sur l'ordinateur. Ceci permet au PC-1600 de fonctionner en tant que calculatrice. Ce mode est également utilisé pour dérouler ou exécuter les programmes engendrés dans le mode PROGRAMME ou chargés dans la mémoire de l'ordinateur à partir d'un équipement de stockage externe.

PROGRAMME Le mode PROGRAMME est utilisé pour la création, l'édition et le listage des programmes. Pour exécuter le programme, vous devez ensuite commuter au mode RUN.

RESERVE Le mode RESERVE est utilisé pour la création et l'édition des chaînes de caractères assignées pour les touches de fonction programmables. Il y a trois modes de menus disponibles, I, II et III, dans le mode RESERVE.

Quel que soit le mode que vous avez sélectionné, lorsque vous mettez l'ordinateur hors tension puis sous tension, l'ordinateur restera dans le même mode que celui qui était en vigueur lors de la mise hors tension. Son symbole sera affiché dans la ligne d'état en haut de l'écran du PC-1600.

Assurez-vous de bien procéder à une double vérification du symbole de mode dans la ligne d'état si vous avez quelques difficultés pendant la manipulation de votre PC-1600. Si vous n'avez pas spécifié le mode correct pour les opérations que vous désirez exécuter, l'ordinateur renverra un code d'erreur sur l'écran. Comme ces messages utilisent un code numérique pour identifier les problèmes, vérifiez la liste des codes d'erreur dans l'Annexe F, pour trouver la signification de l'erreur.

Fonctions des touches d'édition

Cette section décrit brièvement dans les paragraphes suivants de ce chapitre, l'utilisation des touches d'édition pour commander le curseur et procéder à des corrections simples en mode RUN quand vous réglez l'heure et la date et que vous effectuez des calculs.

Les fonctions avancées, utilisant d'autres combinaisons de touches, sont couvertes dans la description du mode EDIT de la section de Référence du langage BASIC.

Voici les touches et fonctions que vous utiliserez dans ce chapitre :

DEL
◄

Touche du curseur avec flèche vers la gauche

C'est l'une des deux touches du curseur pour le déplacement sur une ligne. Quand vous pressez cette touche, le curseur se déplace d'un caractère sur la gauche. Si cette touche est pressée et maintenue enfoncée, le curseur continuera à se déplacer jusqu'à ce qu'il atteigne le début de la ligne logique sur la gauche de l'écran. Utilisée seule, cette touche n'efface pas les caractères, mais fait passer le curseur en-dessous d'eux.

DEL
SHIFT + ◄

Touche d'effacement

Quand vous pressez cette combinaison de touches, le caractère ou l'espace au-dessus de la position du curseur est effacé et tous les caractères situés à droite du curseur, se déplacent d'un caractère vers la gauche. La touche SHIFT ne fonctionne que pour un seul caractère. Vous devez presser la touche SHIFT chaque fois que vous désirez effacer un caractère.

INS
►

Touche du curseur avec flèche vers la droite

Une pression sur cette touche déplace le curseur d'un caractère sur la droite. Si vous pressez et maintenez cette touche enfoncée, le curseur continuera à se déplacer vers la fin de la ligne physique sur l'écran et descendra se positionner au commencement de la ligne suivante .à gauche de l'écran. Comme sa touche de contrepartie, la touche de la flèche gauche, cette touche seule ne peut effacer les caractères, mais fait passer le curseur en-dessous d'eux.

INS
SHIFT + ►

Touche d'insertion

Lorsque vous pressez cette combinaison de touches, un espace est inséré à l'emplacement du curseur et tous les caractères à partir du curseur se déplacent d'un caractère vers la droite. La touche SHIFT ne fonctionne que pour un seul caractère. Vous devez presser la touche SHIFT + INS ► chaque fois que vous désirez insérer un caractère.

BS

Touche d'espacement arrière

La touche QE| (espacement arrière) déplace le curseur vers le caractère de gauche, efface le caractère ou l'espace situé à cet emplacement et rapproche tous les caractères suivants jusqu'à la position du curseur.

Touche de nettoyage

CL

Une pression sur la touche CL (nettoyage) efface la ligne complète où le curseur est positionné et affiche le caractère de sollicitation du système au début de la ligne en tant que signal pour que vous commenciez à taper à nouveau. Le curseur réapparaît après le premier caractère que vous tapez.

Mode INSERT/OWERWRITE (Mode insertion/superposition d'écriture)

CTRL + A

La combinaison de ces touches de service commute à volonté les modes d'édition de texte INSERT ou OVERWRITE. La valeur par défaut à la mise sous tension est en mode OVERWRITE dans lequel les caractères à partir de la position du curseur vers la droite sont effacés tandis que vous tapez par-dessus. Lorsque vous appelez le mode INSERT, les caractères rajoutés seront "insérés" dans la ligne à la position précédent le curseur. Tous les caractères à droite du curseur seront déplacés en conséquence pour faire place aux caractères insérés. Notez que le curseur clignote plus rapidement lorsque le mode INSERT est spécifié.

Voici quelques points à garder à l'esprit lorsque vous avez des problèmes en tapant ou en éditant :

Les fonctions secondaires qui sont imprimées dans les cases de l'ordinateur au-dessus des touches elles-mêmes ne sont effectives que lorsque la touche SHIFT est tout d'abord enfoncée. Donc, si vous oubliez par erreur d'enfoncer la touche SHIFT, par exemple avant INS (insertion) ou DEL (effacement, l'utilisation de ces touches ne fera que de déplacer le curseur vers la droite ou vers la gauche. Vérifiez la ligne de statut pour vous assurer de la présence du symbole SHIFT.

La pression sur la touche SHIFT affecte seulement la frappe du caractère suivant. Pour insérer ou effacer une série de caractères (telle qu'un mot complet), vous devez presser la touche SHIFT + DEL pour chaque caractère ou utiliser la touche BS

Une fois que vous commencez à taper, le curseur se positionne de lui-même à l'emplacement suivant après chaque caractère tapé. Ainsi, le caractère que vous vous préparez à taper commence toujours à la position occupée par le curseur. Autrement dit, lorsque vous déplacez le curseur avant un effacement ou une insertion, ces actions commenceront à partir de la nouvelle position du curseur.

•Pour une correction rapide des introductions courtes, repositionnez simplement le curseur au commencement de la partie inutile et retapez par-dessus. Retaper par-dessus efface seulement les caractères aux endroits recouverts. Cette manipulation n'insère ou n'efface automatiquement aucun espace et ne déplace pas d'un caractère le reste de la ligne que vous retapez.

•La pratique de la frappe et de l'édition est la meilleure manière de connaître son clavier. Utilisez les notes de ces pages pour vous rafraîchir la mémoire lorsque vous avez des difficultés.

Réglage de l'heure et de la date

L'horloge en temps réel dans le PC-1600 vous permet d'utiliser les commandes BASIC pour mettre l'ordinateur sous tension à une heure et un jour précis, faire retentir une alarme ou afficher un message sur l'écran.

Cette section vous montre comment régler l'heure et le jour réels. Ultérieurement, vous pouvez afficher l'heure chaque fois que vous le désirez ou que vous voulez écrire des programmes qui impliquent l'heure exacte pour exécuter des opérations.

Les commandes BASIC qui sont utilisées pour les fonctions se rapportant à l'heure sont TIME, TIME$, DATES, WAKE$ et ALARM$. Voyez le Dictionnaire des commandes BASIC pour la description complète de ces commandes.

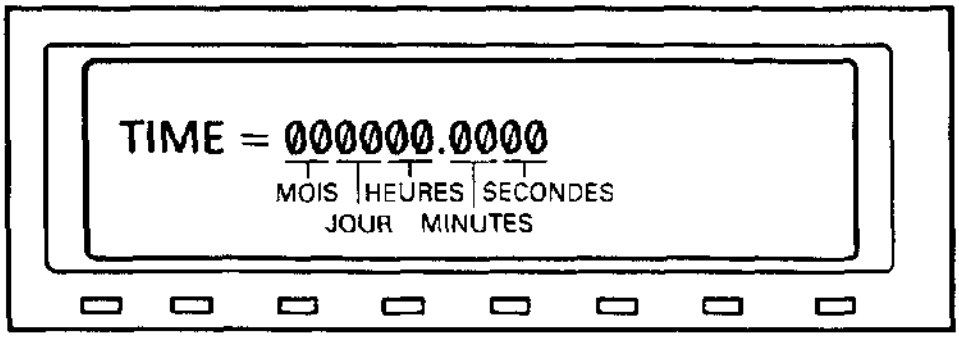
Une de ces commandes, TIME, est utilisée dans les phases suivantes pour régler l'heure et la date à l'origine. Vous utiliserez également cette commande chaque fois que vous voudrez simplement voir l'heure exacte.

Pour régler l'heure et la date réels:

1Avec l'ordinateur sous tension, assurez-vous tout d'abord que le mode d'exploitation est spécifié sur RUN ou PROgramme. Utilisez la touche MODE si nécessaire.

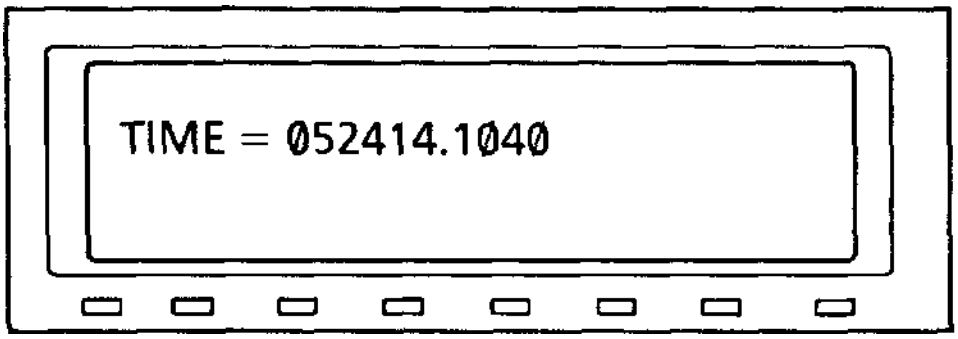
2Quand le caractère de sollicitation du système > apparaît, tapez TIME=. N'utilisez pas d'espaces et n'oubliez pas le signe égal (=). L'ordinateur est maintenant prêt à accepter les chiffres de la date et de l'heure.

3Le format des nombres correspondant à la date et à l'heure est de deux chiffres pour le mois, le jour, l'heure, les minutes et les secondes. Remarquez le point décimal entre les heures et les minutes sur l'illustration.



4Vous devez inclure un 0 avant chaque chiffre pour tous les nombres de 1 à 9 (donc, tapez 01 à 09). N'oubliez pas le point décimal entre les heures et les minutes et ne laissez aucun espace. Utilisez l'affichage officiel à 24 heures où minuit est 00, midi est 12, 6 heures du soir 18, etc.

5L'illustration suivante vous montre un exemple du réglage de l'heure et de la date pour le mois de mai (05), le 24^{e} jour (24), à 2 heures de l'après-midi (14), 10 minutes (10) et 40 secondes (40). L'année ne peut être spécifiée.

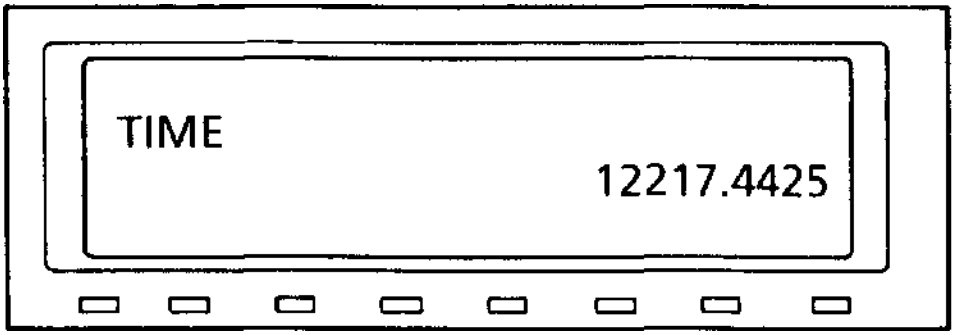


6 Tapez maintenant de 01 à 12 pour le mois, de 01 à 3.1 pour le jour, de 00 à 23 pour les heures, un point pour introduire le point décimal, de 00 à 59 pour les minutes et finalement de 00 à 59 pour les secondes.

7 L'ordinateur affichera l'heure à la seconde où vous pressez la touche **ENTER**. Vous pouvez trouver plus facile de synchroniser la pression de la touche **ENTER** avec le 4^{e} top de l'horloge parlante. Toute autre horloge ou montre suffira si vous n'êtes pas très exigeant sur la précision des secondes.

8 Maintenant, quand vous êtes prêt, pressez la touche **ENTER** pour spécifier l'heure et la date.

9 Toutes les fois que vous désirez lire l'heure, tapez simplement T I M E et pressez la touche **ENTER**. Le signe égal (=) n'est pas nécessaire. Quand vous procédez à cette opération, l'écran se présente d'une manière un peu différente, avec l'heure et la date affichées sur la ligne suivante à l'extrême droite de l'écran. Pour les autres utilisations de l'heure et de l'horloge, voyez les commandes mentionnées dans le Dictionnaire des commandes. Voici à quoi l'écran ressemble quand vous affichez l'heure et la date réelles.


TIME
12217.4425

Calculs sur le PC-1600

5

Ce chapitre vous indique comment exécuter une variété de calculs sur votre ordinateur PC-1600. Avec son clavier numérique séparé, le PC-1600 vous permet de faire des calculs simples rapidement, comme si vous utilisiez une calculatrice de poche. Et, quand l'ordinateur est connecté à l'une des imprimantes en option, CE-1600P ou CE-150 (avec le MODE 1), la commutation d'un interrupteur situé sur l'imprimante vous permet d'imprimer vos calculs et leurs résultats tandis que vous travaillez. Dans ce sens, cette combinaison de votre ordinateur PC-1600 et son imprimante ressemble davantage à une machine à additionner avec une sortie imprimée. Egalement à des niveaux avancés, les opérations de fonctions du langage de programmation BASIC incorporé dans l'ordinateur vous permettent d'effectuer des calculs dans les programmes.

Si vous faites des fautes de frappe, reportez-vous à la section Fonctions des touches d'édition du chapitre 4.

Spécification du mode

Spécifiez le mode RUN pour utiliser l'ordinateur d'après ce chapitre. Lorsque vous écrirez ultérieurement vos programmes en utilisant les fonctions de calculs du PC-1600 en BASIC, vous spécifierez le mode PROgramme. Voyez la section de Référence du langage BASIC pour ce niveau avancé d'opérations.

Vérifiez la ligne d'état au sommet de l'écran pour vous assurer que le symbole RUN est bien affiché. Pressez si nécessaire la touche MODE pour spécifier le mode RUN.

Fonctions des touches

Cette section introduit l'utilisation des touches numériques et décrit leurs fonctions et caractéristiques spéciales.

Touche ENTER

ENTER

La touche ENTER, comme vous l'avez appris dans les sections précédentes, est un signal à l'ordinateur qu'une séquence d'introduction est complète et qu'il doit prendre l'action ou l'exécution de la donnée qu'il a reçue. Dans les calculs simples, la touche ENTER est utilisée à la place du signe "égal", ce qui signifie, retourner le résultat de l'expression. Ne pressez pas la touche = (égal) pour exécuter des calculs car cette fonction de touche est réservée à un usage spécial en BASIC.

Touche astérisque

*

La touche astérisque renvoie la fonction de multiplication, habituellement indiquée par le symbole arithmétique x.

Touche barre oblique

/

La touche barre oblique renvoie la fonction de division, indi quée autrement par le symbole arithmétique -s-.

e Touche E

La touche E de caractère est utilisée pour introduire les nombres en représentation scientifique avec les exposants

Exemples de calculs

Voici quelques exemples de calculs sur le PC-1600. Ils montrent la représentation conventionnelle des expressions, les touches nécessaires pour les introduire dans l'ordinateur et les résultats affichés sur l'écran. Essayez-les sur votre PC-1600 pour vous familiariser avec les touches et leurs fonctions. Le dernier exemple montre comment les opérations en BASIC incorporé fonctionnent avec l'ordinateur. Après avoir exécuté ces calculs, essayez-en quelques-uns des vôtres. Notez qu'après qu'une équation est exécutée et son résultat affiché, l'équation suivante que vous introduisez apparaît sur la ligne suivante tandis que l'écran roule vers le haut.

Exemple 1

2 + 3x4 = ?

Pressez : ENTER 2 + 3*4

Résultat: 14

Exemple 2

36 -î- (1 + 2) = ?

Pressez : ENTER

36/(1 +	2)
Résultat	12

Exemple 3

$$5 \times 10^3 \div (4 \times 10^{-3}) = ?$$

Pressez: **ENTER**

Résultat: 1250000

Exemple 4

$\sin^2 30° = ?$

D E G. **ENTER** (SIN 3Ø) ^ 2 **ENTER**

Pressez:

Résultat: 0,25

Calculs en série

Comme vous l'avez remarqué à partir des exemples précédents, l'expression que vous introduisez apparaît sur la ligne supérieure de l'écran. Quand vous pressez la touche **ENTER** le résultat est affiché sur la droite, à la fin de la ligne suivante.

Vous pouvez utiliser ce résultat en tant que première introduction du calcul suivant en pressant une touche de fonction numérique, sans retaper ce résultat.

L'illustration ci-dessous montre ce qui se produit sur l'écran lorsque vous enchaînez les résultats en des calculs en série. Notez que vous n'avez pas besoin de retaper le résultat, il apparaît automatiquement au commencement de la ligne suivante tandis que vous introduisez les fonctions numériques. Lorsque vous atteignez la ligne du bas, l'écran roule d'une ligne vers le haut pour laisser un espace pour l'expression ou le résultat suivant.

Essayez d'introduire les mêmes nombres dans l'illustration et regardez ce qui arrive dans votre PC-1600.

Lorsque vous pressez ces touches L'écran se présente comme cela ...

Pressez:

12/6 **ENTER**

*5 + 10 ENTER

+ 30 ENTER

Fonction de rappel

Dans le chapitre 4 vous avez appris comment utiliser les touches fléchées à droite et à gauche pour déplacer le curseur et éditer vos entrées. Il y a deux autres usages pratiques de ces touches durant les calculs directs,

La fonction de rappel vous permet d'afficher à nouveau et de revenir à une équation exécutée pour réintroduire les valeurs ou faire d'autres changements.
Cette caractéristique peut également être utilisée d'une manière légèrement différente après qu'un code ERROR est affiché. Vous trouverez plus d'informations sur les codes ERROR dans la section suivante.

Vous pouvez revenir à tout moment à une équation après avoir vu le résultat, et apporter quelques changements dans les entrées. Ceci peut être pratique pour essayer quelques modifications "WHAT IF" simples pour voir quels effets ont les nouvelles valeurs sur les résultats des calculs. Ou, quand les résultats produits ne semblent pas corrects, il se peut qu'une valeur erronée ait été introduite à un point de l'équation et nécessite une correction.

Après que vous avez pressé la touche ENTER pour exécuter un calcul, le résultat est affiché à la fin de la ligne suivante. Si vous désirez revenir à l'équation elle-même, pressez la touche ▶ (flèche à droite) pour afficher à nouveau l'équation avec son curseur positionné sous le premier caractère. Autrement, la pression de la touche ◀ (flèche à gauche) positionne le curseur sous le dernier caractère.

Vous pouvez alors déplacer le curseur à l'aide des touches ◀ ou ▶ à la position que vous désirez et suivre les procédures que vous avez apprises dans la section sur les Fonctions des touches d'édition, dans le chapitre 4.

Vous avez maintenant apporté les changements nécessaires pour afficher à nouveau l'équation, pressez ENTER pour l'exécuter à nouveau. Cette fois, le résultat sera affiché sur la ligne après la ligne sur lequel le curseur était durant l'édition. Le résultat se superposera à une partie de l'équation occupant la ligne.

Codes d'erreurs

Tandis que l'ordinateur essaye d'exécuter l'équation que vous avez introduite, il va tout d'abord vérifier que l'introduction de l'expression est bien validée. Si votre entrée est incorrecte, l'ordinateur n'exécutera pas le calcul, mais affichera un code ERROR sur la ligne suivante, à la place du résultat.

Tout d'abord, vérifiez la liste des codes ERROR dans l'Annexe F pour trouver sa signification et vérifiez également l'équation elle-même.

Une erreur typique très simple est l'omission des parenthèses ( ) dans un calcul composé. Si un code d'erreur est affiché, vous devez procéder au vidage avant de continuer à utiliser l'ordinateur.

Choisissez une de ces deux manières pour traiter une erreur :

- En tant que variation de l'utilisation des touches fléchées de la précédente section sur la Fonction de rappel, pressez ◀ ou soit ▶ vider le code ERROR et afficher à nouveau l'équation. Cette fois le curseur sera positionné au point où l'erreur se trouve.

  Vous pouvez alors corriger l'erreur et ré-exécuter le calcul en pressant la touche ENTER

  Ou si, après avoir vérifié l'équation, vous désirez la réintroduire complètement, une pression sur n'importe laquelle des touches suivantes videra le code ERROR comme l'équation de l'écran et de la mémoire de l'ordinateur.
  Rappelez-vous que vous devrez alors introduire l'équation à nouveau. Ceci peut être un inconvénient pour les équations longues et compliquées, aussi, décidez pour quelle procédure vous désirez opter. Si vous désirez la réintroduire, pressez :

  ON ou CL ou SHIFT + CL

  Puis tapez l'équation à nouveau et exécutez-la en pressant la touche ENTER.

Les codes ERROR dont il est question dans ce chapitre concernent la validité ou l'inconsistance des introductions durant les calculs. Rappelez-vous que le PC-1600 peut afficher au même moment d'autres codes ERROR pour des raisons diverses telles que l'avertissement de l'épuisement des piles. Vérifiez toujours la liste du code des erreurs pour en trouver la signification avant d'entreprendre quoi que ce soit.

Opérations des fonctions BASIC

Votre PC-1600 dispose d'un nombre de fonctions incorporées qui apportent une réduction des calculs en mode direct comme dans les programmes. Deux de ces fonctions, Sinus et Degré, ont été brièvement introduites dans les exemples de calculs. La liste qui suit montre toutes les fonctions qui sont disponibles dans le BASIC de l'ordinateur. Pour une description complète de chacune d'elles, voyez le Dictionnaire des commandes BASIC, dans la partie IV de ce manuel. Vérifiez aussi les chapitres se rapportant à la section de Référence du langage BASIC pour plus de détails sur l'utilisation de ces expressions et opérateurs et sur l'évaluation prioritaire des expressions arithmétiques.

Les fonctions suivantes sont incorporées dans le BASIC de votre PC-1600:

ABS	Valeur absolue	LN	Logarithme neperieu
ACS	Arc cosinus	LOG	Logarithme décimal
ASN	Arc sinus	PI	Valeur de pi
ATN	Arc tangente	RND	Nombre aléatoire
COS	Cosinus	SGN	Signe
DEG	Décimal/degré	SIN	Sinus
DMS	Degré/décimal	SQR	Racine carrée
EXP	Exposant	TAN	Tangente
INT	Nombre entier		

Les valeurs introduites pour les fonctions trigonométriques peuvent être des valeurs en degrés, radians ou gradients selon la spécification de DEG, RAD ou GRAD.

UTILISATION DES INTERFACES ET DES EQUIPEMENTS E/S EN OPTION

La clé de la flexibilité de votre ordinateur de poche PC-1600 est son aptitude à s'élargir en un système à caractéristiques complètes, compact, utilisant une variété équipements E/S en option.

Les sections de cette partie offrent une description introductive des équipements individuels, des références aux commandes BASIC utilisées pour les faire fonctionner et des notes spéciales sur les connexions et la gestion des erreurs.

Le chapitre 7 est une description de la compatibilité des équipements périphériques de l'ordinateur PC-1500 qui peuvent être utilisés avec le PC-1600.

6 Elargissement du PC-1600

Ce chapitre vous indique comment constituer votre système à partir de votre ordinateur de poche PC-1600, en utilisant les nombreux équipements en option de SHARP. Vous pouvez utiliser ces adjonctions pour créer le système compact décrit dans les pages suivantes.

Vue d'ensemble du système

Votre ordinateur SHARP PC-1600 renferme une grande puissance et de nombreuses caractéristiques pour sa taille. Il est même encore plus universel quand il est raccordé à un ensemble d'équipements en option spécialement conçus, aussi bien qu'aux options compatibles du PC-1500. A partir de votre ordinateur PC-1600, vous pouvez constituer un système compact pour manipuler une variété de besoins de traitement des données, de stockage et de sorties.

Cette section du manuel vous présente brièvement les options disponibles et mentionne les points-clés que vous devez garder à l'esprit en utilisant ces équipements. Pour des explications détaillées, référez-vous toujours aux manuels de fonctionnement accompagnant ces équipements individuels.

L'ordinateur lui-même est caractérisé par trois ports d'interfaces standards : série RS-232C, série optique et entrée analogique. Ces ports sont utilisés pour connecter l'ordinateur directement sur des équipements externes comme les imprimantes, modems ou détecteurs.

L'addition des modules de RAM dans l'un ou les deux logements au dos du PC-1600 vous permet d'élargir la mémoire interne de 16 K à un maximum de 80 K et, avec le choix d'un module de programme, vous pouvez écrire et stocker vos programmes et les connecter dans l'ordinateur selon vos besoins.

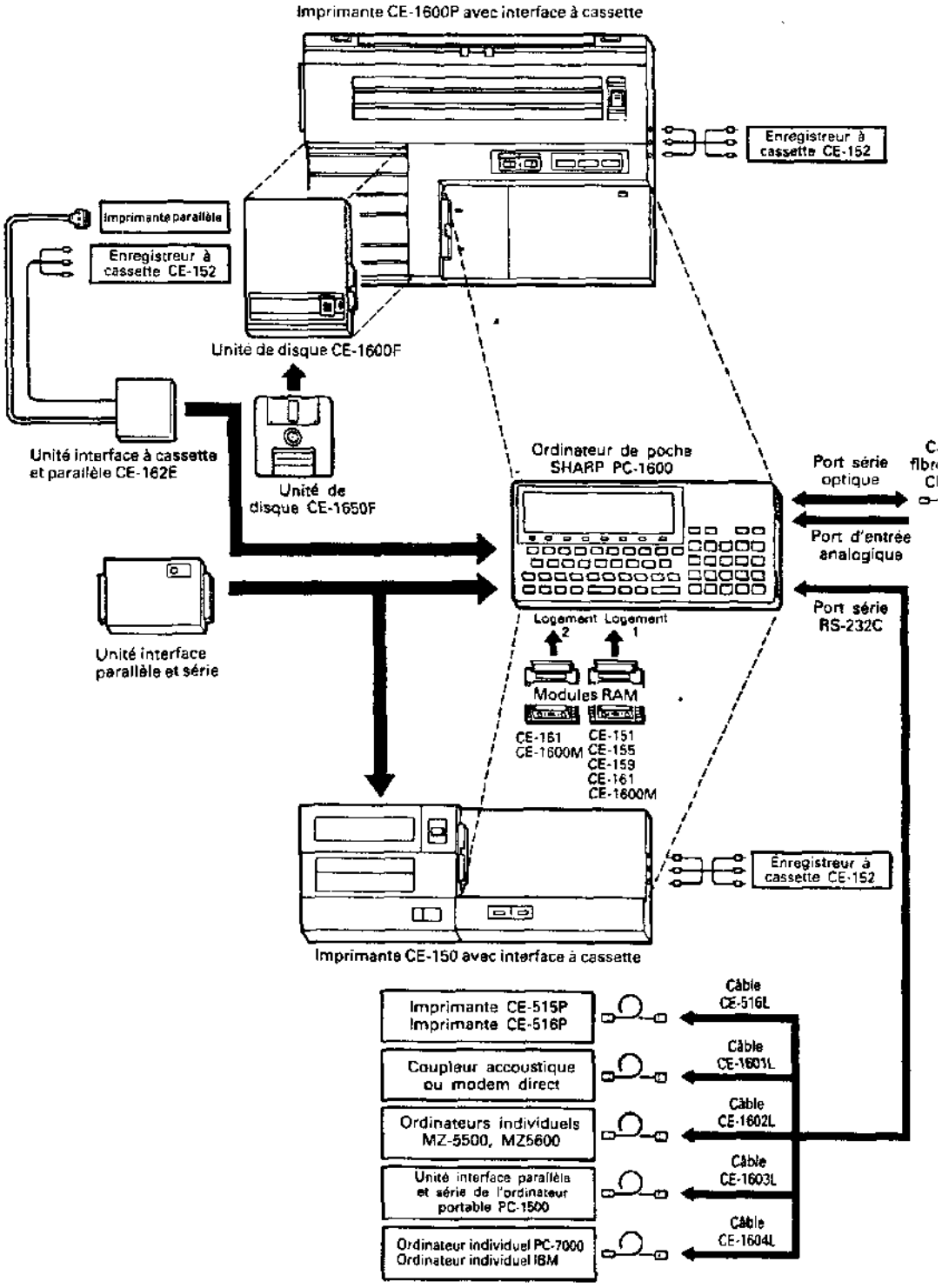
L'ordinateur PC-1600 est conçu pour une connexion directe sur l'imprimante CE-1600P en option pour les sorties sur papier des programmes, des textes et des graphiques en quatre couleurs. L'unité intégrée, ordinateur et imprimante, fournit la base sur laquelle les autres options viennent se connecter.

L'unité de disque CE-1600F est un une adjonction utile pour un grand stockage des programmes et données et se monte directement sur le corps de l'imprimante pour donner forme à une configuration de base de votre système compact.

Une autre caractéristique clé de votre ordinateur PC-1600 est sa compatibilité avec l'ordinateur de poche SHARP PC-1500. Cela signifie qu'il ne partage pas seulement les programmes et commandes BASIC du PC-1500, mais aussi qu'il est connectable

sur ses équipements E/S en option. Ces derniers comprennent un enregistreur à cassette, des modules de RAM, une imprimante et des interfaces parallèle et série.

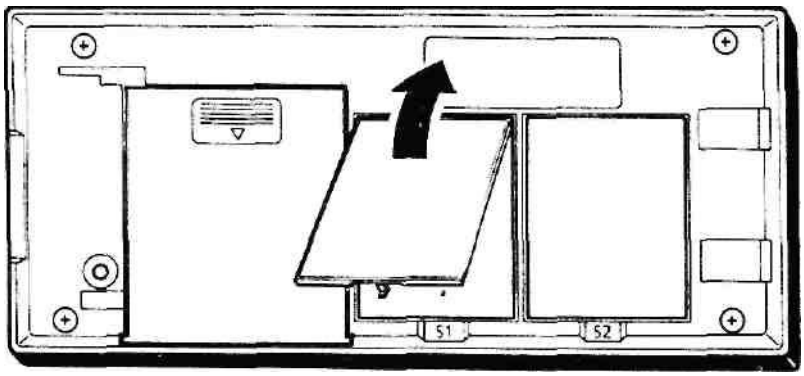
Cette illustration montre la variété des connexions possibles pour constituer votre système d'ordinateur de poche. Les équipements compatibles du PC-1500 sont décrits dans le chapitre 7, Utilisation des périphériques du PC-1500.



Configuration du système ordinateur de poche SHARP PC-1600

Module d'expansion de RAM

Vous pouvez élargir la mémoire interne RAM de 16 K de votre PC-1600 en enfichant des modules RAM dans les deux logements situés dessous l'ordinateur. Cette section décrit les types et l'utilisation de ces modules, donne les points-clés pour la connexion et l'utilisation et liste les commandes BASIC et les codes ERROR s'y rapportant.



LOGEMENT 1 LOGEMENT 2

Logements 1 et 2 des modules dessous le PC-1600

Les modules RAM sont disponibles sous deux formes : les modules de mémoire, qui sont des mémoires d'extension pour la RAM incorporée de l'ordinateur et les modules de programmes qui ont une touche de protection en écriture et une pile de sauvegarde interne. Les modules de programmes vous permettent d'utiliser la mémoire en tant que mémoire de programme pour écrire et stocker les programmes que vous pouvez connecter sur l'ordinateur quand c'est nécessaire. Une des caractéristiques des modules de programmes CE-161 et CE-1600M est qu'ils peuvent être utilisés en tant que disque RAM sur lesquels vous pouvez écrire et lire des fichiers de données comme sur un disque.

Le tableau suivant compare les caractéristiques des modules RAM disponibles pour votre ordinateur PC-1600.

	CE-151	CE-155	CE-159	CE-161	CE-16QQM
Type de module:					
Module mémoire	●	●			
Module programme			●	●	●
Capacité Utilisation:	4K	8K	8K	16K	32K
Mémoire d'extension					
Mémoire de programme	●	●	●	●	●
Disque de RAM			●	●	●
Sauvegarde par pile Touche protection écriture Utilisé en logement 1/2				●	●
			●	●	●
			●	●	●
	1	1	1	1/2	1/2

Points-clés

1 Mise hors tension Mettez toujours l'ordinateur hors tension avant d'installer ou de remplacer les modules de RAM.
Quand l'ordinateur est hors tension, la caractéristique sauvegarde de mémoire protège le contenu de la mémoire des modules d'extension de RAM aussi bien que de la RAM interne de l'ordinateur.

2Mise sous tension N'enlevez jamais un module de RAM quand l'ordinateur est sous tension. Lorsque vous mettez l'ordinateur sous tension après avoir monté un module de RAM pour la mémoire d'extension, l'ordinateur affiche un message vous rappelant de vider et de réinitialiser la zone mémoire modifiée. Ce message n'est pas affiché pour les modules de programmes utilisés en mémoire de programme ou disque de RAM. Reportez-vous à la section Mise sous tension du chapitre 3.

3Logements des modules II y a deux logements de modules de RAM en dessous de l'ordinateur. Notez que vous devez utiliser le logement 1 quand vous ne montez qu'un module de RAM pour la mémoire d'extension. Reportez-vous à la section Mise sous tension du chapitre 3.

4Manipulation des modules Ne touchez jamais les connecteurs sur les modules ou à l'intérieur des logements. L'électricité statique présente un danger réel pour la mémoire de l'ordinateur. Dans les pièces moquettée et quand l'hiver est sec, assurez-vous de bien de bien libérer la charge statique en touchant le bouton d'une porte ou tout objet similaire en métal avant de manipuler les modules.

5 Touche protection d'écriture Les modules de programme ont une touche protection en écriture pour éviter les changements des données ou programmes dans la mémoire.
Positionner cette touche sur OFF avant d'utiliser un module de programme pour la mémoire d'extension.

6Sauvegarde sur pile Les modules de programme ont des piles au lithium pour préserver les données et les programme en mémoire quand le module est enlevé de l'ordinateur. Les modules de mémoire qui ne disposent plus de cette pile de sauvegarde perdent le contenu de la mémoire quand le module est remplacé.

7Commande IIMIT Les modules de RAM peuvent être utilisés de trois manières : les modules de mémoire pour élargir la zone mémoire de l'utilisateur et les modules de programmes en tant que mémoire programme exclusive ou disque RAM. Utilisez la commande INIT en BASIC pour désigner l'utilisation de ce module.

8 Pour référence

Référez-vous au manuel de fonctionnement du module pour toute information sur:

Les instructions détaillées sur l'utilisation et les soins à apporter; les spécifications; le remplacement de la pile dans les modules de programmes.

Voyez ce manuel de fonctionnement pour les informations sur:

Les commandes se rapportant au BASIC; les conditions de mise sous ou hors tension; les logements des modules; les notes de remplacement.

Commandes des fichiers sur disque de RAM

Les commandes suivantes sont pour utiliser un module de programme comme disque de RAM. Notez que ces commandes sont les mêmes que les commandes de fichiers utilisées pour les disques. Notez que TITLE, la dernière commande de la liste, est utilisée pour sélectionner un module de programme mais ne peut être utilisée quand le module est un disque de RAM. Pour une description complète de chaque commande, reportez-vous au Dictionnaire des commandes BASIC à la PARTIE IV de ce manuel.

Commande	Description
BLOAD	Charge le programme langage machine du disque de RAM en mémoire.
BSAVE	Sauvegarde le programme langage machine de la mémoire au disque de RAM.
CLOSE	Termine l'accès à un fichier.
COPY	Copie un fichier vers ou d'un disque de RAM et un équipement E/S externe.
EOF	Renvoie la valeur indiquant la fin du fichier durant une introduction à partir d'un fichier en série.
FILES	Affiche les informations de répertoire pour les fichiers spécifiés.
INIT	Initialise et spécifie l'usage des modules de RAM.
INPUT#	Lit les articles à partir d'un fichier en série.
KILL	Efface un fichier sur le disque de RAM.
LFILES	Liste les informations du répertoire sur des fichiers spécifiés sur disque de RAM vers l'imprimante CE-1600P ou les accès séquentiels.

LOAD/LOAD*	Charge un fichier d'un disque de RAM en mémoire.
LOC	Renvoie le nombre d'enregistrements lus ou écrits depuis l'ouverture du fichier.
LOF	Renvoie la taille du fichier en octets.
MAXFILES	Spécifie un nombre maximum de fichiers qui peuvent être ouverts en même temps.
NAME	Change le nom d'un fichier
OPEN	Valide l'entrée et la sortie des données vers ou en provenance d'un fichier sur disque de RAM.
Print# SAVE/	Inscrit une donnée dans un fichier en série sur disque de RAM.
SAVE*	Sauvegarde les fichiers de données ou de programmes sur disque de RAM.
SET	Met en ou hors fonction la protection de fichier pour les fichiers sur disque de RAM.
TITLE	Sélectionne le module de programme dans le logement 1 ou 2 pour l'introduction ou l'exécution d'un programme. Invalide la commande quand le module de programme est utilisé en tant que disque de RAM.

Codes ERROR

Il s'agit d'une liste de codes ERROR qui se rapporte directement à l'utilisation des modules de RAM. Reportez-vous aussi aux codes ERROR des disques et fichiers dans la section sur l'unité de disque. Les codes des erreurs sont résumés dans l'Annexe F.

Codes	Description
27	Commande illégale. Le périphérique adressé n'est pas connecté.
101	Invalide le nom de l'équipement spécifié par les instructions TITLE ou NEW.
102	Invalide la sélection de l'équipement, (l'équipement n'est pas connecté).
103	La mémoire du module de RAM est pleine. Les paramètres INIT ne peuvent être spécifiés.

Interfaces série

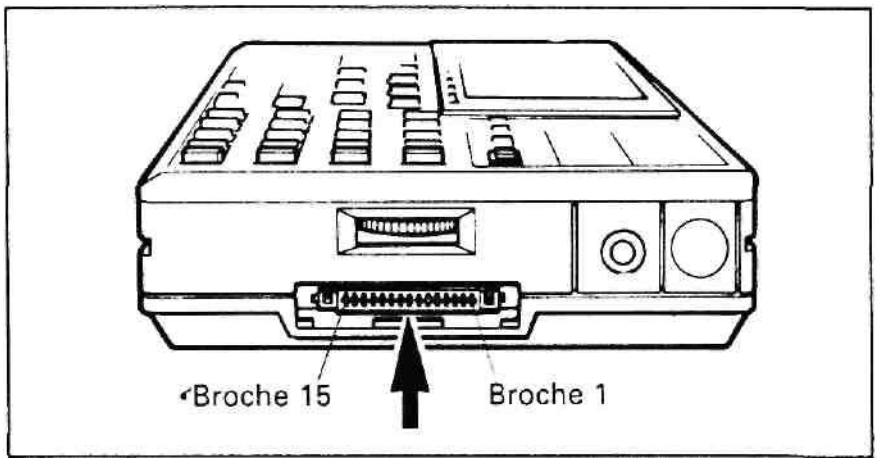
Le PC-1600 est doté de deux interfaces série incorporées :
RS-232C et optique. Les sections suivantes décrivent l'utilisation des port série, donnent les listes des spécifications et des brochages de connecteurs, puis donnent un résumé des commandes BASIC et des codes ERROR s'y rapportant.

Port série RS-232C

Le port série RS-232C répond aux normes ElA pour la transmission des données asynchrones entre votre ordinateur de poche PC-1600 et les autres équipements tels qu'une imprimante série, un modem, ou un autre ordinateur.

En commandes BASIC, le port RS-232C est désigné sous C0M1: et ne peut être utilisé en même temps que le port série optique, COM2 :.

Le port série RS-232C sur votre PC-1600 supporte le signal Cl d'un modem téléphonique pour mettre automatiquement sous tension l'ordinateur, ou recevoir une interruption.



Port série RS-232C

Points-clés

1Mise hors tension Avant d'effectuer les connexions, mettez toujours l'ordinateur et les équipements externes hors tension.

2Manipulation Ne touchez jamais les broches des connecteurs sur l'ordinateur, les câbles ou les équipements externes. Remettez les couvercles des connecteurs quand ceux ci ne sont pas utilisés.

3Protocole de transmission La transmission des données entre cet ordinateur et les autres équipements doit être faite selon des paramètres compatibles. Ces paramètres de transmission sont habituellement spécifiés en commandes BASIC. Voyez la liste résumée dans cette section et la description complète de chaque commande dans le Dictionnaire des commandes BASIC.

4Connexion des équipements Assurez-vous qu'un équipement est bien connecté au port désigné avant de tenter la transmission. Si vous envoyez les données sur un équipement non existant, l'ordinateur se "bloquera". Pressez la touche **ON** (rupture) pour réinitialiser sans perdre le fichier.

5Pour référence :

Référez-vous aux manuels de fonctionnement des équipements connectés (autre ordinateur, imprimante, modem) et les câbles, pour toute information sur :

La communication des paramètres; les spécifications des câbles; les brochages des connecteurs; les procédures de transmission et de réception des données.

• Voyez ce manuel de fonctionnement pour les informations sur :

Les commandes BASIC en relation; les paramètres et protocoles de transmission de votre ordinateur PC-1600; les brochages du connecteur RS-232C; les codes ERROR.

Les valeurs par défauts de l'interface RS-232C

La liste suivante vous donne les valeurs par défaut, ou pré spécifiées des paramètres de transmission pour l'accès RS-232C, COM1:. Ces valeurs peuvent être changées par deux commandes BASIC : SETCOM ou PCONSOLE. Voir le Dictionnaire des commandes BASIC pour les instructions détaillées sur l'utilisation de ces commandes.

Commande	Paramètre	Valeur par défaut
SETCOM		1200
	Longueur de mot (5-8 bits)	8
	Parité (paire, impaire, non)	Non
	Bits d'arrêt (1 ou 2)	1
	Protocole X-ON/OFF	Oui
	Protocole normal/spécial (7 bits de données seulement)	Oui
PCONSOLE	Longueur de ligne	Illimitée
	Fin du code de ligne	CR

Brochages du connecteur RS-232C

Le tableau suivant liste le nombre de broches RS-232C, les noms des signaux et symboles avec leur description.

Broche n°	Signal	Symbole	Description
2	Transmission des données (sortie)	TXD (SD)	Ligne de données en série depuis l'ordinateur.
3	Réception des données (entrée)	RXD (RD)	Ligne de données en série d'un équipement externe à l'ordinateur.
4	Demande de transmission (sortie)	RTS (RS)	Signal d'intervention de l'ordinateur.
5	Remise à zéro pour transmission (entrée)	CTS (CS)	Signal d'intervention d'un équipement extérieur à l'ordinateur.
6	Ensemble des données prêt (entrée)	DSR (DR)	Signal d'intervention d'un équipement extérieur à l'ordinateur.
7	Signal de mise à masse	SG *	Signal d'intervention d'un équi-la pement extérieur à l'ordinateur.
8	Détection de porteuse (entrée)	CD	Signal d'intervention d'un équipement extérieur à l'ordinateur.
9	Indicateur d'appel (entrée)	Cl	Signal d'intervention d'un équipement extérieur à l'ordinateur.
10	Tension logique (sortie)	Vcc	Quand l'ordinateur est ON, la tension logique Vcc (4 à 4,7 V) est continue.
14	Terminal des données (sortie)	DTR (ER)	Signal d'intervention d'un équipement extérieur à l'ordinateur.

NOTES:

1 Les signaux d'intervention de sortie RTS et DTR peuvent être spécifiés rapides ou lents par commande BASIC. Voyez OUTSTAT dans le Dictionnaire des commandes.

2 Les états rapides ou lents des signaux d'intervention d'entrée CTS, DSR, CD et Cl peuvent être lus à l'aide de la commande BASIC INSTAT. Voyez INSTAT dans le Dictionnaire des commandes.

3 Le signal d'intervention d'entrée Cl peut interrompre l'ordinateur quand il est sous tension comme hors tension. Ce signal commute l'ordinateur sous tension quand celui-ci est arrêté. Voyez ON PHONE GOSUB, PHONE ON/OFF/STOP et WAKE$ dans le Dictionnaire des commandes.

Commandes E/S RS-232C

Les commandes suivantes sont pour l'utilisation du port RS-232C, désigné sous COM1: pour les transmissions des données en série. Pour une description complète de chaque commande, reportez-vous au Dictionnaire des commandes BASIC à la partie IV de ce manuel.

Commande	Description
COM$	Renvoie la chaîne des valeurs de paramètres de transmission spécifiées dans l'instruction SETCOM.
COMn ON/OFF/STOP	Valide ou invalide les interruptions reçues.
INIT	Spécifie la taille du tampon receveur. Renvoie les
INSTAT	valeurs des signaux d'intervention.
ON COMn GOSUB	Provoque l'exécution d'un saut à la ligne spécifiée d'un Sous-programme quand l'interruption est reçue.
ON PHONE GOSUB	Provoque l'exécution d'un saut au numéro ligne spécifié quand l'entrée est reçue de la ligne téléphonique.
OUTSTAT	Spécifie l'état rapide ou lent des signaux d'intervention
PCONSOLE	Spécifie la longueur de la ligne et le code EOL.
PHONE ON/OFF/STOP	Valide ou invalide les interruptions reçues du modem télé phonique.
PZONE	Spécifie la zone d'impression pour la sortie avec l'instruction LPRINT.
RCVSTAT	Spécifie le protocole de liaison reçu et la valeur de tempori-sation.
SETCOM	Spécifie le protocole de transmission.
SETDEV	Spécifie le port RS-232C pour l'entrée et la sortie.
SNDBRK	Envoie les codes de rupture pour la transmission à deux entrées.
SNDSTAT	Spécifie le protocole de liaison envoyé et la valeur de temporisation.

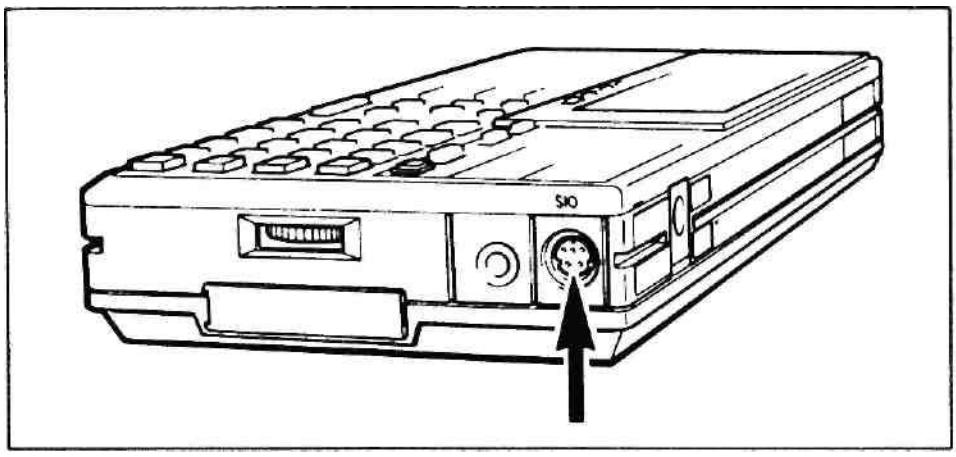
Codes ERROR

Voici la liste des codes ERROR qui se rapportent directement à l'utilisation des deux interfaces série : les ports RS-232C (COM1:) et série optique (COM2:) durant la transmission des données. Les codes ERROR sont résumés dans l'Annexe F.

Code	Description
140	Paramètre invalide spécifié dans l'instruction SETCOM.
141	La taille du tampon receveur spécifiée dans l'instruction INIT est trop large (supérieure à 16383 octets ou à la mémoire disponible).
142	Erreur durant la réception des données (erreur de parité, erreur de dépassement, erreur de bloc et erreur complète de tampon receveur).
143	Erreur de temporisation. Aucune réponse de l'autre partie pour une durée plus longue que la valeur de temps spécifiée dans les instructions RCVSTAT ou SNDSTAT.
144	Le port série spécifié dans l'instruction SETDEV est déjà ouvert.

Port série optique

Votre PC-1600 supporte la transmission des donnée par port série optique vers et en provenance d'un équipement externe via un connecteur à 5 broches, étiqueté SIO, sur la droite de l'ordinateur.



Port série optique (SIO)

Cette section décrit les broches du connecteur, donne un résumé des commandes BASIC et la liste des points-clés pour la connexion et l'utilisation.

L'utilisation du port série optique est identique au port série RS-232C de la norme EIA, mais il apporte des avantages distincts. L'un d'eux est une vitesse de transmission plus grande jusqu'à 38400 bauds en mode semi-duplex pour votre PC-1600.

La transmission optique offre également le moyen d'envoyer et de recevoir des données série dans les environnements adverses car le câble à fibres optiques, CE-1600L optique, permet un transfert des données sans parasitage, isolé électriquement.

En commande BASIC, le port série optique est désigné sous COM2: et ne peut être utilisé en même temps que le port RS-232C, C0M1 :.

Points clés

1Mise hors tension Avant d'effectuer les connexions, mettez toujours l'ordinateur et les équipement externes hors tension.

2Manipulation Ne touchez jamais les broches des connecteurs sur l'ordinateur, les câbles ou les équipements externes. Remettez les couvercles des connecteurs quand ceuxci ne sont pas utilisés. Les fibres optiques dans le câble sont endommagées quand le câble est plié, provoquant des erreurs de données et des pertes de transmission. S'il n'est pas possible d'éviter cette situation, il est préférable d'enrouler le câble plutôt que de le plier.

3Protocole de transmission La transmission des données entre cet ordinateur et les autres équipements doit être faite selon des paramètres compatibles. Ces paramètres de transmission sont habituellement spécifiés en commandes BASIC. Voyez la liste résumée dans cette section et la description complète de chaque commande dans le Dictionnaire des commandes BASIC.

4Connexion des équipements Assurez-vous qu'un équipement est bien connecté au port désigné avant de tenter la transmission. Si vous envoyez les données sur un équipement non existant, l'ordinateur se "bloquera". Pressez la touche **ON** (BREAK) pour réinitialiser sans perdre le fichier.

5 Pour référence:

- Référez-vous aux manuels de fonctionnement des équipements connectés (autre ordinateur, etc.) et les câbles, pour toute information sur:
  La communication des paramètres; les spécifications des câbles; les brochages des connecteurs; les procédures de transmission et de réception des données.

- Voyez ce manuel de fonctionnement pour les informations sur:
  Les commandes BASIC en relation; les paramètres et protocoles de transmission de votre ordinateur PC-1600; les brochages du connecteur RS-232C; les codes ERROR.

Les valeurs par défauts du port série optique

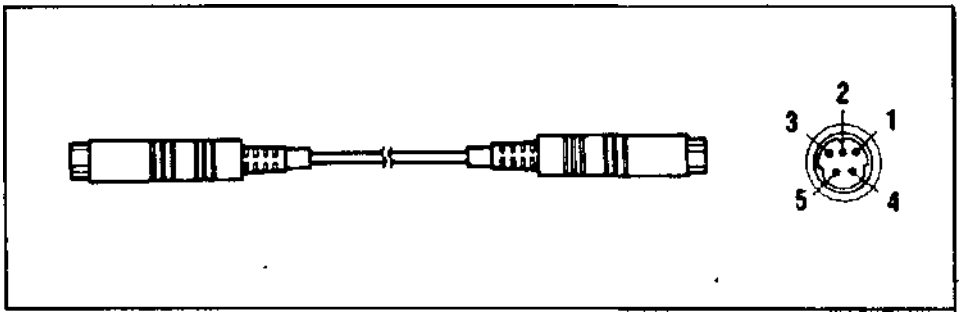
La liste suivante vous donne les valeurs par défaut, ou pré spécifiées des paramètres de transmission pour le port série optique COM2:. Ces valeurs peuvent être changées par deux commandes BASIC : SETCOM ou PCONSOLE.

Voir le Dictionnaire des commandes BASIC pour les instructions détaillées sur l'utilisation de ces commandes.

Commande	Paramètre	Valeur par défaut
SETCOM	Débit binaire (bps)	38400
	Longueur de mot (5-8 bits)	7
	Parité (paire, impaire, non)	Paire
	Bits d'arrêt (1 ou 2)	2
	Protocole X-ON/OFF	Oui
	Protocole normal/spécial (7 bits de données seulement)	Oui
PCONSOLE	Longueur de ligne	Illimitée
	Fin du code de ligne	CR

Brochages du connecteur séquentiel optique

L'illustration suivante montre le câble à fibres optiques CE-1600L et les brochages du connecteur à 5 broches utilisé pour l'accès séquentiel E/S optique. Le tableau suivant liste le nombre de broches RS-232C, les noms des signaux et symboles avec leur description.



Brochage du connecteur et du câble à fibres optiques

Description des brochages et des signaux

Broche n°	Signal	Symbole	Description
1	Réception des (entrée)	RD	Ligne de données série d'un équipe-ment externe à l'ordinateur.
2	Signal de mise à la terre	GND	Mise à la terre habituelle.
3	Emission des (sortie)	SD	Ligne de données série de i'ordina-teur d'un équipement externe.
4	Tension logique	Vcc	Quand l'ordinateur est ON, la tension logique Vcc (4 à 4,7 V) est continue.
5	Tension logique	Vcc	Identique à ci-dessus.

Commandes E/S série optique

Les commandes suivantes sont pour l'utilisation du port série optique COM2: dans les transmissions des données série. Pour une description complète de chacune des commandes, reportez-vous au Dictionnaire des commandes BASIC à la PARTIE IV de ce manuel.

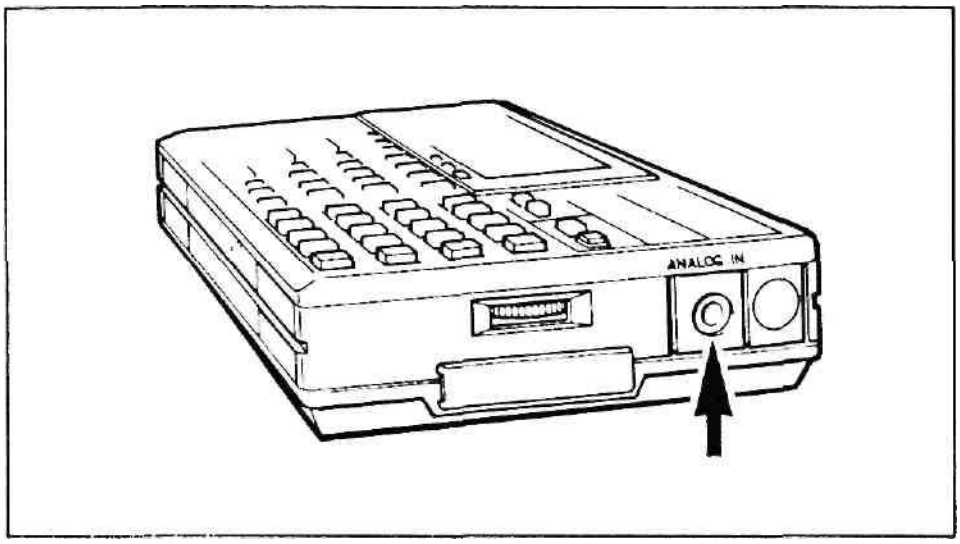
Commande	Description
COM$	Renvoie la chaîne des valeurs de paramètres de transmission spécifiées dans l'instruction SETCOM.
COMn ON/OFF/STOP	Valide ou invalide les interruptions reçues.
INIT	Spécifie la taille du tampon receveur.
ON COMn GOSUB	Provoque l'exécution d'un saut à la ligne spécifiée d'un sous-programme quand l'interruption est reçue.
PCONSOLE	Spécifie la longueur de la ligne et le code EOL.
PZONE	Spécifie la zone d'impression pour la sortie avec l'instruction LPRINT.
RCVSTAT	Spécifie le protocole de liaison reçu et la valeur de temporisation.
SETCOM	Spécifie le protocole de transmission
SETDEV	Spécifie le port RS-232C pour l'entrée et la sortie.
SNDBRK	Envoie les codes de rupture pour la transmission à deux entrées.
SNDSTAT	Spécifie le protocole de liaison envoyé et la valeur de temporisation.

Codes ERROR

La liste des codes ERROR communiquée dans le paragraphe concernant le port RS-232C couvre les erreurs pouvant survenir pour les ports série RS-232C (COM1 :) et optique (COM2:) Voyez cette liste des codes ERROR quand vous utilisez le port série E/S optique pour les transmissions de données. Les codes ERROR sont résumés dans l'Annexe F.

Port d'entrée analogique

Votre ordinateur PC-1600 peut recevoir des signaux analogiques d'équipements externes tels que des capteurs via un port d'entrée analogique, étiquetée ANALOG IN, sur la droite de l'ordinateur.



Port d'entrée analogique (ANALOG IN)

Cette section décrit les spécifications du port d'entrée analogique et les commandes BASIC s'y rapportant.

A l'aide des commandes BASIC, vous pouvez convertir les signaux analogiques dans un niveau numérique équivalent pour enregistrer les mesures en provenance des sources analogiques et pour initialiser l'exécution des sous-programmes dans un programme.

Caractéristiques électriques

L'utilisation effective du port d'entrée analogique dépend de la combinaison de l'équipement de la source du signal analogique et de la commande de logiciel en BASIC (ou langage machine) pour convertir précisément en valeur numérique équivalente. Le tableau suivant liste les spécifications électriques du port analogique.

Spécifications d'entrée analogique

Tension d'entrée maximum	4V
Plage de tension d'entrée	de 0 à 2.495 V
Niveau numérique converti	de 0 à 255 (8 bits) (Toute tension d'entrée supérieure à 2.495 V est convertie à 255.)
Erreur de conversion	3 % ± 2 chiffres
Impédance d'entrée	100 Kohms ± 2%

IMPORTANT: Pour éviter d'endommager l'ordinateur et le port d'entrée analogique, assurez-vous de la tension d'entrée est située dans une plage de 0 à 4V.

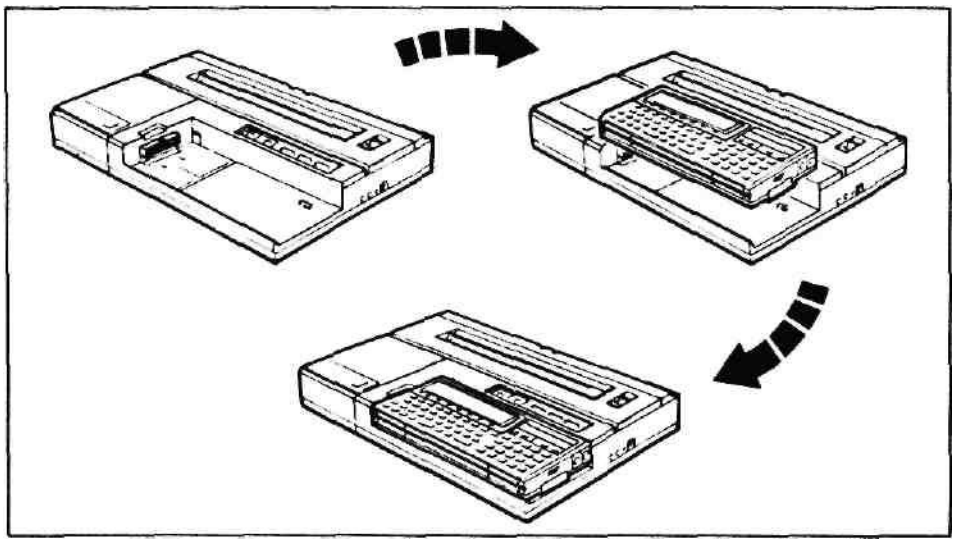
Commandes d'entrée analogique

Les commandes suivantes sont destinées à l'utilisation du port d'entrée analogique pour recevoir des données d'un équipement analogique externe. Pour une description complète des commandes, reportez-vous au Dictionnaire des commandes BASIC à la partie IV de ce manuel.

Commande	Description
ADIN ON/OFF/STOP	Valide et invalide les interruptions reçues.
AIN	Retourne une valeur correspondante pour entrer un niveau de tension analogique.
ON ADIN GOSUB	Provoque l'exécution d'un branchement à un numéro de ligne spécifié quand la tension d'entrée est en dehors de la plage fixée.

Imprimante avec interface à cassette

L'imprimante CE-1600P apporte à votre ordinateur de poche la capacité d'impressions en quatre couleurs des textes et graphiques et est une base sur laquelle vous pouvez connecter d'autres options.



Connexion à l'imprimante CE-1600P

Cette section décrit les points clés pour utiliser l'imprimante et son interface à cassette incorporée et liste les commandes BASIC et les codes ERROR s'y rapportant.

L'imprimante CE-1600P est conçue pour accepter l'ordinateur PC-1600 selon une connexion à glissière pour un profil intégré régulier. L'unité de disque CE-1600F en option est également connectable sur le berceau de l'imprimante de la même manière, pour compléter votre système de base d'ordinateur de poche.. La connexion directe de l'imprimante au bus du système de l'ordinateur et la connexion de l'unité de disque éliminet la nécessité de câbles séparés entre les équipements.

L'imprimante est alimentée par des piles NiCad rechargeables qui alimentent également l'unité de disque et l'ordinateur aussi, si nécessaire. Un adaptateur CA, le EA-160, est un accessoire standard pour opérer l'imprimante à partir d'une prise électrique murale en intérieur ou pour économiser les piles.

Points-clés

1Mise hors tension L'imprimante ne dispose pas d'interrupteur de mise sous et hors tension. Lorsqu'elle est connectée à l'ordinateur, l'imprimante est alimentée ou non quand l'ordinateur est mis sous/hors tension. Avant d'effectuer les connexions ou les déconnexions, mettez toujours l'ordinateur hors tension.

2Manipulation Ne touchez jamais les broches du connecteur de l'imprimante ou le bus de système de l'ordinateur. Lorsque l'électricité statique devient un problème, éliminez-là en prenant en main un objet métallique. Replacez les couvercles des connecteurs lorsqu'ils ne sont pas utilisés.

3Piles déficientes Utilisez un adaptateur CA pour recharger les piles de l'imprimante à l'utilisation initiale. Votre ordinateur PC-1600 dispose de trois moyens pour vous prévenir que les piles sont déficientes : le symbole BATT sur la ligne d'état de l'affichage, un message CHECK lors de la mise sous tension et les codes ERROR durant l'exécution.

4Bouton RESET Lorsque l'ordinateur est connecté à l'imprimante, le bouton RESET situé sous l'ordinateur n'est plus accessible. Un bouton RESET est équipé au dos du berceau de l'imprimante pour remplir la même fonction. Reportez-vous au chapitre 3, Réinitialisation de l'ordinateur.

5 Interface à cassette L'interface à cassette incorporée permet la connexion d'un enregistreur de données tel que l'enregistreur à cassette CE-152 SHARP en option. La commande à distance sur l'imprimante vous permet de contrôler la puissance de l'enregistreur par une commande BASIC. Voyez la section suivante, Utilisation d'un enregistreur à cassette.

6 Pour référence

Référez-vous au manuel de fonctionnement de l'imprimante CE-1600P pour des informations sur:

Détails d'utilisation et de soins; remplacement des stylos, mise en place du papier et réglages; spécifications générales et listes des accessoires.

Voyez ce manuel de fonctionnement pour les informations sur:

Les commandes BASIC se rapportant à l'imprimante; les codes ERROR; spécification des modes graphique et texte.

Commandes de l'imprimante

Les commandes suivantes sont utilisées pour écrire les données sur l'imprimante, spécifier diverses valeurs de format d'impression et spécifier les modes d'impres-sin. Pour une description complère de chaque commande, reportez-vous au Dictionnaire des commandes BASIC à la PARTIE IV de ce manuel.

Commande	Description
COLOR	Spécifie la couleur du stylo.
CSIZE	Spécifie la dimension des caractères pour l'impression.
GLCURSOR	Déplace le stylo à une position spécifiée en mode graphique.
GRAPH	Spécifie l'imprimante en mode graphique.
LCURSOR	Déplace le stylo à une colonne spécifiée en mode texte.
LF	Alimente le papier le nombre spécifié de lignes.
LLINE	Tire une ligne ou une série de segments de lignes entre des coordonnées absolues.
LLIST/LLIST*	Liste les lignes de programme à l'imprimante
LPRINT	Sort les données pour l'imprimante.
PAPER	Spécifie la gamme d'impression verticale et le type de papier.
PCONSOLE	Spécifie le format d'impression et le code EOL pour les données à l'imprimante.
PITCH	Spécifie l'espacement des caractères et la densité verticale des lignes en mode texte.
PZONE	Spécifie la zone d'impression pour la sortie avec l'instruction LPRINT.
RLINE	Tire une ligne ou une série de segments de ligne-entre des coordonnées relatives.

ROTATE	Spécifie l'orientation d'impression des caractères.
SORGN	Spécifie la position du stylo réel en tant qu'origine.
TAB	Déplace le stylo jusqu'à la colonne spécifiée dans le mode texte
TEST	Exécute un essai d'impression en quatre couleurs.
TEXT	Spécifie l'imprimante en mode texte.

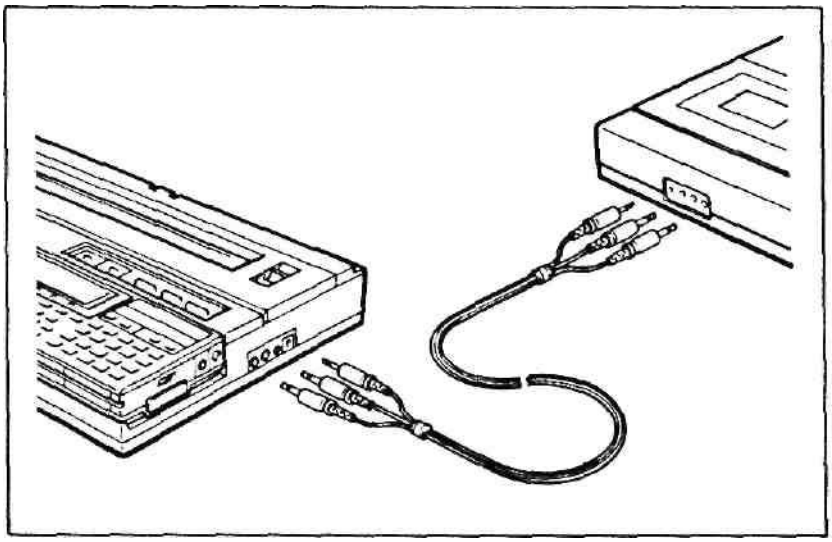
Codes ERROR

Voici une liste des codes ERROR qui est en relation directe avec l'utilisation de l'imprimante. Pour le résumé général des codes ERROR, voyez l'Annexe F.

Code	Description
70	Le stylo est hors gamme X=-2048, Y-2047
71	La forme inverse alimente plus de 10,24 cm dans le mode texte (CE-150 seulement).
72	Spécification incorrecte du paramètre TAB.
73	Commande interdite dans le mode réel. Commande graphique utilisée en mode texte et vice-versa.
74	Trop de paramètres dans les instructions LLINE et RLINE.
76	Essai de LLIST (lister) une ligne de programme plus longue que la valeur PCONSOLE de 17 ou moins.
78	Piles déficientes, stylo non en place, ou imprimante bloquée. Ne peut exécuter les commandes LPRINT ou LLINE.
79	Signaux couleurs non sortis sur l'imprimante (CE-150 seulement).
80	Piles déficientes, imprimante bloquée.

Utilisation d'un enregistreur à cassette

Vous pouvez connecter et exploiter un enregistreur des données, tel que l'enregistreur à cassette SHARP CE-152, via l'interface sur l'imprimante CE-1600P. L'illustration suivante montre la connexion du câble sur l'imprimante comme sur l'enregistreur. Cette section décrit les commandes BASIC pour l'écriture et la lecture des données vers et en provenance de la bande, et liste les codes ERROR et les points-clés pour la connexion et l'utilisation.



Connexion d'un enregistreur à cassette sur l'imprimante CE-1600P

Pomts-cles

1Mise hors tension Assurez-vous de bien mettre l'ordinateur et l'enregistreur hors tension avant d'effectuer la connexion des câbles.

2Connexion des câbles L'imprimante CE-1600P dispose de trois jacks sur son côte gauche pour connecter les câbles en provenance d'un enregistreur à cassette. Les jacks sont étiquetés : REM (Remote) pour commander à distance de l'alimentation de l'enregistreur, MIC (Microphone) pour l'écriture, ou l'enregistrement des données de l'ordinateur sur bande, et EAR (Earphone) pour la lecture, ou reproduction des données de la bande vers l'ordinateur. Le câble qui vient avec l'enregistreur à cassette SHARP CE-152 est doté de codes colorés pour faciliter les connexions. Les mini fiches qui s'adaptent dans les jacks EAR et MIC sont de même dimension. Vérifiez que vous n'avez pas interverti leurs branchements si vous avez des problèmes.

3Commande à distance Vérifiez que le commutateur REMOTE sur l'imprimante est bien sur ON si vous voulez pouvoir mettre l'enregistreur sous/hors tension à l'aide de la commande BASIC RMT ON/OFF.

4Les enregistreurs à cassette autres que SHARP Les enregistreurs à cassette autres que le SHARP CE-152 peuvent avoir une sortie incompatible avec l'ordinateur PC-1600. En tant qu'information sur le problème, le code ERROR 44 est alors affiché durant une tentative de lecture des données en provenance de la bande. Pour permettre à l'ordinateur de lire à *partir de la* bande avec l'enregistreur connecté, introduisez la commande suivante:

POKE&F1A9, 0ËZHH3

IMPORTANT: Faites très attention lorsque vous entrez la commande parce que vous effectuez des changements dans les données du système. Si vous faites une erreur de frappe et pressez la touche **ENTER**, *vous pouvez exécuter une réinitialisation ALL RESET de l'ordinateur, ce qui remettrait à zéro toutes les mémoires des programmes et données réels.*

Pour restaurer l'ordinateur dans sa capacité d'utilisation du CE-152, introduisez la commande suivante:

POKE & F1A9, 128 **ENTER**

Si vous ne restaurez pas l'ordinateur et essayez d'utiliser l'enregistreur CE-152, le codes ERROR 44 sera affiché à nouveau.

5 Pour référence:

- Référez-vous aux manuels de fonctionnement de l'imprimante CE-1600P, de l'enregistreur CE-152 et des autres enregistreurs de données pour des informa tions sur :
  Les connexions et câbles de l'enregistreur, les spécifications, l'entretien et les soins d'ensemble et les procédures d'exploitation.
- Voyez ce manuel pour les informations sur:
  L'utilisation des commandes BASIC pour écrire et lire les données sur et de la bande; les codes ERROR.

Commande E/S de la bande de cassette

Voici une liste des commandes BASIC que vous pouvez utiliser pour lire et écrire des données sur et de la bande, vérifiez l'exactitude des données enregistrées sur la bande et mettez l'enregistreur sous/hors tension à distance, à partir de l'ordinateur. Pour une description détaillée de chaque commande, voyez le Dictionnaire des commandes à la PARTIE IV de ce manuel.

Commande	Description
BLOAD	Charge le programme langage machine de la bande en mémoire.
BSAVE	Sauvegarde le programme langage machine de la mémoire sur la bande.
CHAIN	Permet à un programme BASIC de charger et exécuter un autre programme sur la bande.
CLOAD	Charge en mémoire un programme ou un programme à usage de touche de fonction à partir de la bande.

CLOAD?	Vérifie l'absence d'erreur d'enregistrement du programme sur la bande.
CLOADM	Charge le programme langage machine en mémoire à partir de la bande.
CLOSE	Termine l'accès à un fichier.
COPY	Copie un fichier sur ou depuis une bande et un équipement E/S externe.
CSAVE	Sauvegarde le programme entier ou en partie de la mémoire sur la bande.
CSAVEM	Sauvegarde le langage machine de la mémoire sur la bande.
EOF	Retourne la valeur indiquant la fin de fichier durant l'introduction d'un fichier séquentiel sur la bande.
INPUT#	Lit les items en provenance d'un fichier séquentiel sur la bande.
LOAD/LOAD*	Charge un fichier de la bande en mémoire.
MAXFILES	Spécifie un nombre maximum de fichiers qui peuvent être ouverts en même temps.
MERGE	Fusionne le programme sauvegardé sur la bande avec un programme en mémoire.
OPEN	Valide l'entrée et la sortie des données vers ou en provenance d'un fichier sur la bande.
PRINT#	Ecrit les données sur un fichier à sortie séquentielle sur la bande.
RMT ON/OFF	Valide ou invalide la télécommande d'alimentation de l'enregistreur à cassette.
SAVE/SAVE#	Sauvegarde les fichiers de données ou de programmes sur la bande.

Codes ERROR

Voici une liste des codes ERROR qui se rapportent directement à l'utilisation de l'enregistreur à cassette pour stocker et charger les données. Pour une description complète de tous les codes ERROR, consultez l'Annexe F.

Code	Description
40	Erreur de syntaxe dans la commande de bande.
42	Mémoire insuffisante pour charger le programme.

43 Erreur de vérification de bande. Après avoir exécuté CLOAD?, la copie en mémoire ne correspond pas à l'original de la bande. Erreur de lecture de bande possible. Chargez et vérifiez à nouveau.

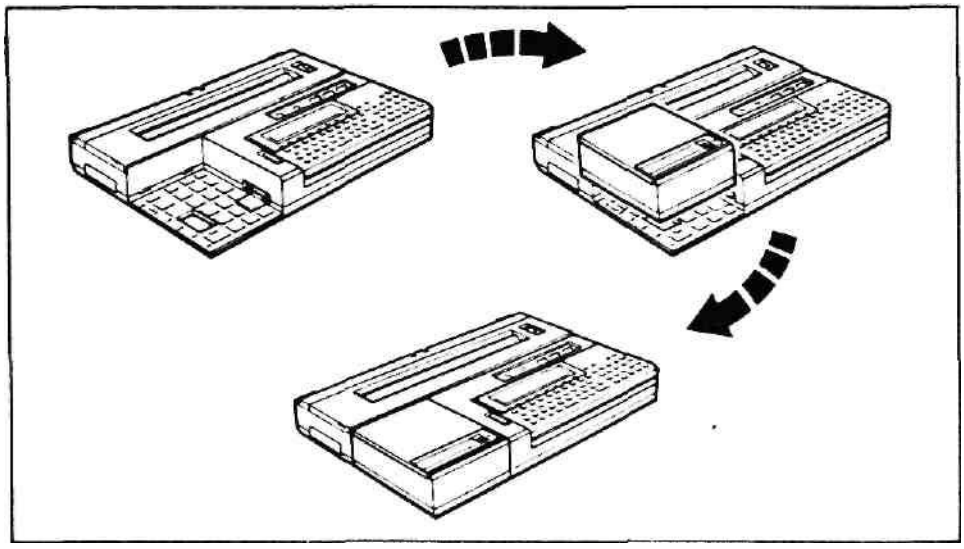
La frappe ne correspond pas à l'instruction INPUT#. Le type de variable et le type de donnée ne sont pas compatibles. Essayez de lire une donnée alphanumérique en donnée numérique ou vice-versa.

Sortie incompatible de l'enregistreur à cassette ou erreur de lecture de bande.

44

Unité de disque

L'addition de l'unité de disque CE-1600F en option à votre combinaison ordinateur-imprimante complète le système de base de votre ordinateur de poche.



Connexion d'une unité de disque CE-1600F sur l'imprimante CE-1600P

Cette section décrit les points clés pour la connexion et l'utilisation de l'unité de disque et liste les commandes BASIC et les codes ERROR s'y rapportant.

Les disques de poche CE-1650F à double face utilisés dans l'unité peuvent stocker jusqu'à 61 K de données sur chaque face, faisant d'eux un outil efficace pour travailler sur un grand volume de données qui limite l'utilisation de la bande ou des modules programme de RAM. Même si les disques sont dans un petit format de 2V2", leur large capacité de stockage et les commandes BASIC similaires les rendent aussi faciles à utiliser que ceux des formats plus grands.

L'unité de disque elle-même est connectée directement sur l'imprimante, éliminant ainsi les câbles. Lorsque l'ordinateur et l'unité de disque sont connectés sur l'imprimante, l'unité complète a alors une apparence intégrée, d'un seul bloc.

Points clés

1Mise hors tension Assurez-vous de bien mettre l'ordinateur sous/hors tension avant d'effectuer la connexion ou la déconnexion de l'unité de disque. Ne touchez jamais les broches des connecteurs et remettez toujours les couvercles des connecteurs lorsque ceux-ci ne sont pas utilisés.

2Formatage des disques Les nouveaux disques doivent être mis en format. Les disques sont à double face et chaque face doit être mise en format séparemment. La face du disque qui fait face est celle qui est sollicitée. Le disque doit être éjecté puis retourné pour que l'autre face puisse être sollicitée. Pour mettre le disque en format, tapez :

INIT"X:" et pressez ENTER.

Le formatage des disques prend moins d'une minute par face. Quand le témoin lumineux vert s'éteint, éjectez le disque, retournez-le et répétez la même opération sur l'autre face.

3Manipulation du disque Tandis que le disque est sollicité, un témoin lumineux vert s'allume en haut de l'unité. Ne jamais éjecter le disque pendant ce moment, car la rotation de la tête de l'unité peut endommager le disque et toutes les données stockées dessus, sans oublier le mécanisme lui-même. De plus, n'éjectez pas un disque quand un fichier est ouvert.

4Alimentation L'unité de disque n'est pas dotée d'une alimentation autonome et son courant est fourni par les piles rechargeables de l'imprimante. Si les piles déficientes font s'allumer le témoin d'avertissement BATT sur la ligne d'état de l'écran, n'utilisez plus l'ordinateur et connectez l'adaptateur CA pour le rechargement ou directement. Une utilisation prolongée de l'unité de disque après que le témoin BATT est allumé, peut engendrer des erreurs et des pertes de données.

5Fichiers Les disques CE-1650F peuvent stocker jusqu'à 48 fichiers, ou 61 K, sur chaque face. Tous les fichiers sont fermés quand l'ordinateur est mis hors tension, manuellement en pressant sur la touche OFF ou en envoyant l'instruction POWER en BASIC.

6Disques de sauvegarde La version du BASIC utilisée sur votre ordinateur de poche SHARP PC-1600 reconnaît les noms de deux aiguillages pour l'unité de disque, X:, Y:. Le nom par défaut est X:, mais vous pouvez spécifier l'autre. A l'aide de ces deux noms, vous pouvez facilement copier les disques contenant des programmes et données importants. Notez que vous devez copier les fichiers un par un et spécifier le nom du fichier d'extension. Le nom du fichier d'extension* ne peut être utilisé pour spécifier tous les fichiers sur disque pour la copie. Supposons que vous ayez un disque dans l'unité et que vous l'avez spécifié sur X:. Pour copier un fichier de ce disque sur un autre disque, tapez la commande suivante :

COPY "X: <nom du fichier. * >" TO "Y: <nom du fichier. * >"

Quand vous pressez la touche ENTER, cette commande dit à l'ordinateur de copier le fichier spécifié sur le disque de X: sur le disque de Y:. Comme X: et Y: peuvent être utilisés dans la même unité de disque, l'ordinateur copiera le fichier du disque réel en mémoire, puis vous demandera d'insérer le nouveau disque par la phrase de sollicitation suivante :

Set diskette for Y: (ou vice-versa)

Lorsque vous voyez ce message, éjectez le disque utilisé, insérez le nouveau disque et pressez ENTER. Le fichier spécifié sur le disque d'origine est maintenant copié de la mémoire sur le nouveau disque de sauvegarde. Quand un fichier est complètement copié, l'unité de disque cesse de fonctionner, le témoin vert s'éteint et le caractère de sollicitation > apparaît sur l'écran. Continuez à copier les autres fichiers à partir du disque d'origine. Quand vous avez terminé, vous pouvez éjecter le disque de sauvegarde, le mettre de côté et continuer à utiliser l'ordinateur. Voyez COPY et FILES dans le Dictionnaire des commandes BASIC pour plus d'information sur la copie des fichiers.

7 Pour référence:

- Référez-vous aux manuels de fonctionnement de l'imprimante CE-1600P et de l'unité de disque CE-1600F pour les informations sur:
  La connexion de l'unité; l'insertion, l'éjection des disques; la protection de l'écriture; les spécifications; les procédures d'exploitation et les soins généraux.

- Voyez ce manuel de fonctionnement pour les informations sur:
  Les commandes. BASIC pour écrire et lire des données sur disque; les codes ERROR; la création et l'édition des fichiers en BASIC.

Commandes E/S de disque

Voici la liste des commandes BASIC qui sont utilisées pour lire et écrire des fichiers à partir du et sur le disque. Pour une description détaillée de chaque commande, voyez le Dictionnaire des commandes BASIC à la PARTIE IV de ce manuel.

Commandes des fichiers de disque

Commande	Description
BLOAD	Charge du disque en mémoire le programme langage machine.
BSAVE	Sauvegarde le langage machine de la mémoire sur le disque.
CLOSE	Termine l'accès à un fichier.
COPY	Copie un fichier sur depuis un disque et un équipement E/S externe.
EOF	Retourne la valeur indiquant la fin de fichier durant l'introduction d'un fichier séquentiel sur la bande.
FILES	Affiche les informations de répertoire pour les fichiers spécifiés.
INIT	Initialise et met en format un disque vierge.
INPUT #	Lit les items en provenance d'un fichier séquentiel sur la bande.
KILL	Efface un fichier sur le disque.
LFILES	Liste les informations de répertoire sur un fichier spécifié sur disque à l'imprimante ou port série.
LOAD/LOAD*	Charge un fichier du disque en mémoire.
LOC	Renvoie le nombre d'enregistrement écrits ou lus depuis que le fichier a été ouvert.
LOF	Renvoie la taille d'un fichier en octets.
MAXFILES	Spécifie le nombre maximum de fichiers qui peuvent être ouverts en même temps.
NAME	Change le nom d'un fichier.
OPEN	Valide l'entrée et la sortie des données vers ou en provenance d'un fichier sur disque.
PRINT#	Ecrit les données pour un fichier séquentiel sur disque.
SAVE/SAVE*	Sauvegarde les fichiers des programmes ou des données sur disque.
SET	Valide et invalide la protection des fichiers sur disque.

Codes ERROR

Voici une liste des codes ERROR qui se rapportent directement à l'utilisation d'une unité de disque pour stocker et charger les données. Pour une description complète de tous les codes ERROR, consultez l'Annexe F.

Code	Description
150	Trop de fichiers spécifiés dans l'instruction MAXFILES.
151	Le fichier existe déjà. Utilisez un autre nom de fichier.
152	Fichier non trouvé. Vérifiez encore la spécification du nom du fichier.
153	Numéro de fichier incorrect dans l'instruction lecture ou écriture de fichiers. Le fichier spécifié n'est pas encore ouvert.
154	Le fichier est déjà ouvert. Fermez-le et ouvrez-le à nouveau dans un autre mode.
155	Nom d'unité de disque interdit, ou unité de disque non connectée.
156	Spécification de paramètre incorrecte dans l'instruction SET.
157	Nom de fichier interdit, ou spécification incorrecte.
158	Commande non soutenue. Commande invalide pour l'unité de disque.
159	Tentative pour écrire sur un disque à l'écriture protégée.
160	Le disque n'est pas dans l'unité de disque spécifiée.
161	Le disque n'est pas mis en forme avec l'instruction INIT.
162	Erreur de lecture ou d'écriture de disque.
163	Mauvais disque dans l'unité de disque. Les disques ont été changés quand le fichier était ouvert.
164	Erreur de disque plein.
165	Fin du fichier atteinte avec l'instruction du numéro INPUT. Toutes les données ont été lues.
166	Mémoire insuffisante pour la zone de travail du disque. L'espace libre n'est pas suffisant dans la mémoire interne de l'ordinateur pour le système E/S de l'unité de disque.
167	Erreur fatale au disque. Tout le contenu est détruit ou altéré.
168	Pile déficientes sur l'imprimante ou faute du matériel de l'unité de disque.

7 Utilisation des périphériques du PC-1500

La clé de la flexibilité de votre PC-1600 est sa compatibilité avec l'ordinateur de" poche SHARP PC-1500. Cela signifie qu'il peut partager la plus grande partie des équipements périphériques du PC-1500. Ces options sont rapidement décrites dans les pages qui suivent. Consultez votre concessionnaire SHARP pour la disponibilité et les conseils sur leur fonctionnement. Tous les équipements en option sont pourvus de manuels. Lors de l'utilisation de ces équipements avec votre PC-1600, vérifiez bien tous les détails dans les manuels.

Dans ce manuel, nous avons déjà fait référence à certaines de ces options, les modules de RAM et l'enregistreur à cassette. Notez que votre PC-1600 peut fonctionner comme le PC-1500 lorsqu'il est spécifié en MODE 1. Pour toute information complémentaire, voyez le chapitre 9 dans la section de Référence du langage BASIC.

Consultez aussi l'Annexe H sur la compatibilité des commandes BASIC et des périphériques entre le PC-1600 et le PC-1500.

Voice une liste des périphériques du PC-1500 que vous pouvez utiliser avec votre PC-1600.

CE-150	Imprimante avec interface à cassette Une version plus petite de l'imprimante CE-1600P; Offre une impression des textes et des graphiques à quatre couleurs; l'interface à cassette permet la connexion de deux enregistreurs à cassette.
CE-151	Module de RAM Module de mémoire de 4 k pour élargir la zone mémoire de l'utilisateur.
CE-152	Enregistreur à cassette Se connecte sur les interfaces à cassette des imprimantes CE-1600P ou CE-150, ou sur les unités d'interfaces parallèle ou à cassette CE-162E.
CE-153	Clavier de logiciel Permet l'assignement des touches de fonctions pour une entrée plus facile des introductions clavier répétitives du bloc de contact de matrice 10 x 14.

CE-155 Module de RAM

Module de mémoire de 8 k pour élargir la zone mémoire de l'utilisateur.

- CE-158 Unité d'interface série-parallèle

  Fournit l'interface série RS-232C pour la transmission avec les équipements externes tels qu'un autre ordinateur, modem ou imprimante. L'interface parallèle Centronics permet la connexion d'une imprimante parallèle.

- CE-159 Module de RAM

  Module de mémoire de 8 k avec commutateur de protection écriture et pile de sauvegarde.

- CE-161 Module de RAM

  Module de mémoire de 16 k avec commutateur de protection écriture et pile de sauvegarde.

- CE-162E Unité interface à cassette et parallèle

  Permet la connexion d'une imprimante parallèle et d'un enregistreur des données.

- EA-160 Adaptateur CA

  Accessoire standard avec l'imprimante CE-1600P.

- EA-150 Adaptateur CA

  Accessoire standard avec l'imprimante'CE-150.

PARTIE IV

SECTION DE REFERENCE DU LANGAGE BASIC

Cette partie décrit, sous tous ses aspects, la version spécialement adaptée du langage BASIC de SHARP, qui est indispensable pour une polyvalence incomparable de l'ordinateur de poche PC-1600.

Le chapitre 8 présente certains éléments fondamentaux de programmation. Le chapitre 9 donne un aperçu des différents modes dans lesquels le PC-1600 fonctionne en BASIC. Le chapitre 10 concerne le concept de traitement général des données et les fonctions disponibles.

Le chapitre 11 concerne les techniques plus avancées de gestion de fichiers. Le chapitre 12 couvre une communication sérielle avec d'autres dispositifs. Le chapitre 13 explique certaines des techniques de mise au point plus simples.

Le chapitre 14 énumère toutes les commandes et fonctions en langage BASIC dans un ordre alphabétique.

8 Concept de base de programmation

Le programme est une série logique d'instructions qui régnent sur le fonctionnement d'un ordinateur. Ces instructions doivent être écrites en un langage que l'ordinateur est conçu pour comprendre.

Au niveau le plus complexe, le PC-1600 comprend un langage appelé le "code machine". Toutefois, grâce au "traducteur" intégré dans le PC-1600, il est possible de dialoguer avec un langage plus compréhensible pour les utilisateurs et de convertir, d'une façon automatique et interne, les instructions en code machine pour son propre usage. Il est aussi possible, pour les utilisateurs avancés, de programmer l'ordinateur directement par le code machine.

Pour la plupart des gens, le BASIC est cependant un langage de programmation. Le BASIC comprend environ 200 instructions individuelles référencées en tant que commandes, instructions ou fonctions. Les règles de grammaire (ponctuation et structure de phrase) régnant sur la combinaison des instructions individuelles sont appelées "syntaxe".

Introduction des instructions en BASIC

Il existe deux modes d'introduction pour la plupart des instructions BASIC; le mode direct et le mode indirect ou programme. En mode direct, les commandes sont introduites dans l'ordinateur par clavier et exécutées immédiatement après que la touche ENTER ait été enfoncée. Le résultat d'exécution est immédiat. A l'issue de l'exécution de la commande, l'ordinateur se trouve prêt à recevoir une autre commande directe.

Ce mode est similaire au mode de calcul direct décrit dans le chapitre 5 et utile pour un calcul simple non itératif. Il est aussi utilisé pour délivrer des instructions de commande aux dispositifs périphériques, tels l'unité de disques. Certaines commandes ne sont pas disponibles en mode direct. De même, ce mode ne permet d'introduire qu'une seule commande en une fois. Si plus d'une commande est introduite avant d'appuyer sur la touche ENTER, une erreur en résulte (code d'erreur 1) ou, dans certains cas, seule la première commande est acceptée.

En mode programme, les commandes sont introduites ensemble de façon à former quelques lignes, chacune desquelles est affectée, à sa tête, par un numéro dans l'ordre selon lequel elles doivent être exécutées. Ces lignes de programme sont stockées dans l'ordinateur en tant que programme, mais non exécutées avant que l'utilisateur ne spécifie la commande RUN. Une fois la commande RUN.

Spécifiée, l'ordinateur commence l'exécution par la ligne de programme d'un numéro le plus jeune, passe ensuite aux lignes suivantes dans l'ordre et continue de cette sorte jusqu'à ce qu'il n'y ait plus de lignes restant à exécuter.

L'encombrement d'un programme, c'est-à-dire le nombre de lignes de programme, est limité seulement par la capacité de la mémoire de votre ordinateur. Certaines commandes ne sont pas disponibles en mode programme.

Voici un simple programme en BASIC dont chacune des lignes est numérotée par palier de 10. C'est une bonne méthode de programmation, car elle permet, si nécessaire, d'insérer ultérieurement des lignes supplémentaires entre les lignes existantes.

```
10 PRINT "CE PROGRAMME EST UN EXEMPLE"
20 PRINT "PROGRAMME BASIC A EXPLIQUER"
30 PRINT "L'UTILISATION DES NUMEROS DE LIGNE"
40 LET A = 2
50 LET B = 4
60 LET C = A + B
70 PRINT C
80 END
```

La seule différence entre le mode direct et le mode indirect ou programme réside en pratique dans le fait qu'en mode programme, toutes les commandes et instructions doivent être précédées d'un numéro de ligne. L'ordinateur exécute automatiquement la commutation entre les deux modes selon qu'il y a ou non un numéro de ligne.

Cependant, pour exécuter un programme introduit en mode programme, l'ordinateur doit être réglé au mode RUN par la touche MODE. L'ordinateur signale, par l'affichage d'un caractère de sollicitation (>) au début de la ligne sur l'écran, qu'il est prêt à accepter une tâche prochaine soit en mode direct, soit en mode programme.

Exécution d'un programme

Les programmes ne peuvent être exécutés que lorsque l'ordinateur est sur le mode RUN. Plusieurs méthodes sont disponibles pour l'exécution d'un programme qui réside dans la mémoire de l'ordinateur. La méthode la plus courante est de spécifier la commande RUN. Lorsque cette commande est introduite et que la touche ENTER est enfoncée, l'ordinateur commence l'exécution du programme par sa ligne dotée d'un numéro le plus jeune dans la mémoire.

Pour le programme donné à titre d'exemple, l'exécution commence par la ligne 10 et passe à 20, 30 et ainsi de suite, jusqu'à l'instruction END de la ligne 80.

Introduisez les lignes de programme de 10 à 80 en mode PRO à partir du clavier et déroulez le programme en permutant le mode en RUN. A noter que si, durant la programmation, vous appuyez sur la touche ENTER à la fin de chaque ligne, l'ordinateur insère une colonne après le numéro de ligne.

D'autres méthodes sont disponibles pour commencer l'exécution d'un programme par une ligne spécifique ou reprendre l'exécution après l'arrêt de l'ordinateur pour une raison quelconque (voir la description des commandes CONT, GOTO et ARUN sur le Dictionnaire des commandes).

Etiquettes de programme et touche DEF:

Une autre façon très souple de commencer l'exécution du programme est d'utiliser la touche DEF située à gauche des touches de fonction, en haut du clavier. Cette touche permet d'exécuter un programme à partir de l'étiquette positionnée à sa première ligne. L'étiquette doit être soit une lettre majuscule correspondant aux touches des deux rangées de fond du clavier (ASDFGHJKLZXCVBNM), soit =, ou SPACE.

La ligne de programme dotée d'une étiquette apparaît comme suit: 10:

"A"

A noter que l'étiquette A est mise entre guillements et vient juste après le numéro de ligne.

Le programme qui commence par une étiquette à la première ligne, peut être exécuté en appuyant d'abord sur la touche DEF et puis sur la touche correcte de l'étiquette. Pour le programme donné à titre d'exemple, introduisez à nouveau le numéro de ligne 10 à partir du clavier. Pour cela, remettez d'abord l'ordinateur en mode PRO, puis entrez:

10: "A":PRINT "CE PROGRAMME EST UN EXEMPLE" ENTER

Lorsque vous appuyez sur la touche ENTER, la nouvelle ligne 10 remplace automatiquement la ligne 10 existante. Maintenant, rétablissez le mode RUN ei exécutez le programme en appuyant sur les touches DEF + A.

Cette méthode d'exécution d'un programme permet de retenir plusieurs programmes dans la mémoire de l'ordinateur, ces programmes ayant chacun leui propre étiquette à la première ligne. Un programme particulier peut être exécuté en spécifiant correctement son étiquette. Les étiquettes peuvent aussi être utilisées dans de nombreuses commandes en BASIC pour obtenir rapidement les lignes de programme spécifiques. Pour de plus amples détails au sujet dt l'utilisation de l'étiquette, se référer à la description réservée aux commandes LIST, DELETE et MERGE dans le Dictionnaire des commandes.

Affectation de la mémoire

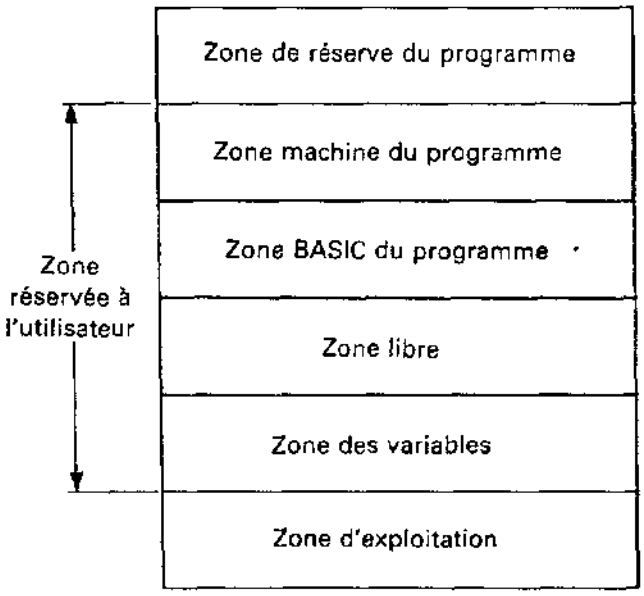
Les zones de la mémoire du PC-1600, mises à la disposition de l'utilisateur pour le stockage et l'exécution des programmes BASIC, peuvent être affectées et supprimées par les commandes BASIC. Les commandes correspondantes sont:

CLEAR, ERASE, MEM, NEW, STATUS, TITLE

Quant à l'affectation de la mémoire, se référer également à l'image mémoire dans l'annexe D.

Zones de mémoire disponibles

La RAM (mémoire vive) intégrée dans le PC-1600 lui-même fournit une zone de 11834 octets, au maximum, à l'utilisateur pour la programmation en BASIC (ou langage machine). Le PC-1600 possède évidemment d'autres zones de mémoire interne qui sont cependant utilisées par le système dont le rôle est de gérer le fonctionnement du PC-1600, de l'unité de disques et des autres dispositifs. Elles ne sont pas normalement disponibles pour l'utilisateur, sauf pour certaines applications très spéciales.



1 Zone réservée à l'utilisateur

La zone réservée à l'utilisateur comprend un espace de stockage des variables utilisées dans un programme et des programmes ayant été chargés dans la mémoire à partir du disque ou de la bande ou introduits par le clavier. S'il n'y a aucun module dans le logement d'extension 1 ou 2, la capacité de la zone réservée à l'utilisateur est de 11834 octets.

Lorsqu'un module de RAM est inséré soit dans le logement 1, soit dans le logement 2, pour l'extension de la mémoire, la zone réservée à l'utilisateur est accrue; la zone interne pour l'utilisateur s'étend dans le module de mémoire. Il en est de même pour le module de programme qui est inséré en utilisant seulement une partie en tant que mémoire extensible.

Afin de trouver à tout moment dans quelle mesure est libre la zone réservée à l'utilisateur, avant ou durant le fonctionnement, utilisez la commande MEM. Cette commande vous informe seulement de l'espace libre restant encore dans la zone réservée à l'utilisateur. Ainsi, l'espace restant non occupé dépend de l'encombrement des programmes stockés dans l'ordinateur même. Si, par exemple, vous avez un module de mémoire de 4K dans le logement 1, la capacité de la zone réservée à l'utilisateur est augmentée de 12K à 12K + 4K = 16K. Dans cette condition, quand un programme BASIC de 3K est chargé dans l'ordinateur à partir du disque, l'espace libre est réduit à 16K - 3K = 13K. La commande MEM fait alors ressortir la capacité restante en octet, à savoir 13000 octets environ.

Lorsque l'unité de disques miniature CE-1600F est reliée à l'ordinateur, une partie de la zone de l'utilisateur est attribuée au système d'entrée/sortie de disques. Il en résulte que la zone réservée à l'utilisateur pour le stockage des programmes est réduite à 10810 octets.

2 Zone de stockage des variables

Tant qu'il y a un programme dans la zone réservée à l'utilisateur, il est impossible de calculer exactement sa capacité restante du fait qu'une certaine partie de la mémoire est utilisée pour le stockage des variables du programme et que sa taille dépend du programme lui-même. Toutefois, si vous désirez vider la zone affectée pour le stockage de ces variables (après avoir exécuté, par exemple, un programme qui utilise un grand nombre de rangées), spécifiez la commande CLEAR. Pour les détails, référez-vous au Dictionnaire des commandes.

3 Zone pour le langage machine

Une partie de la zone réservée à l'utilisateur peut aussi être affectée aux programmes spéciaux en langage machine. Ceci ne concerne cependant que les utilisateurs avancés; la taille de la partie correspondante est définie par la commande NEW. Si vous tapez NEW0 après la remise à l'état initial de l'ordinateur, la zone réservée à l'utilisateur est totalement réinitialisée et aucun espace n'est attribué aux programmes en langage machine.
Dans ce cas, la commande MEM doit indiquer une zone de l'utilisateur de 11834 octets, exactement.

4 Zone pour l'exploitation du système

La zone d'exploitation n'est pas accessible à l'utilisateur. C'est la zone dans laquelle l'ordinateur suit les étapes d'exécution spécifiées par votre programme.

5 Zone de réserve pour le programme

Cette zone est utilisée pour le stockage des contenus de chaque touche de fonction programmable. Lorsque l'ordinateur est mis en mode RESERVE par les touches SHIFT + MODE, les programmes spécifiés par les touches de fonction sont directement chargés dans cette zone spéciale ou peuvent être sauvegardés dans un dispositif depuis cette zone. La taille de la zone de réserve est fixe.

Fichiers en exécution automatique

La fonction Auto-Run (exécution automatique) permet au langage BASIC, sauvegardé sur le disque de l'unité CE-1600F ou sur le disque de RAM dans le PC-1600, d'être exécuté automatiquement lors de la mise sous tension. Afin qu'un programme puisse être exécuté automatiquement à l'aide de cette fonction, il doit être sauvegardé sous le nom de AUTORUN.BAS. En ce qui concerne les renseignements détaillés sur les noms des fichiers, référez-vous au chapitre 11 -Fichiers.

Un programme peut également être exécuté automatiquement à l'aide de la commande ARUN en BASIC (voir page 135). La différence entre les deux fonctions est que pour ARUN, le programme doit déjà être dans la mémoire interne du PC-1600 et peut se voir attribuer n'importe quel nom, tandis que pour la fonction AUTORUN, le programme doit être sur disque ou sur le disque de RAM et doit s'appeler AUTORUN.BAS.

Afin que votre PC-1600 puisse permettre l'exécution automatique des fichiers, procédez de la manière suivante :

1.Sauvegardez un programme sous le nom de AUTORUN. BAS soit sur le disque, soit sur le disque de RAM.
2.A l'aide de la touche MODE, mettez le PC-1600 en mode RUN.
3.Vérifiez que les affichages SMALL et S MODE ne sont pas allumés sur la ligne d'état. Sinon, la fonction Auto-Run ne pourra être lancée.
4.Mettez hors tension comme d'habitude.

Lorsque vous mettrez l'appareil sous tension la fois suivante, ce message s'affichaera :

LOAD"X:AUTORUN.BAS",R (si ce fichier est sur disque)

comme si vous l'aviez tapé vous-même sur le clavier et le fichier sera chargé (voir la commande LOAD page 228) et le programme exécuté automatiquement. Si le fichier est sur disque de RAM, S1 ou S2 sera alors affiché à la place de X :

Différents fichiers AUTORUN peuvent être stockés sur un disque et le disque et le disque de RAM en même temps. Dans ce cas, un fichier AUTORUN sur disque aura priorité sur tous les fichiers AUTORUN sur disque de RAM. De plus, un fichier AUTORUN dans le logement S1: aura priorité sur un fichier AUTORUN dans le logement S2:

Lorqu'un fichier AUTORUN est chargé dans la mémoire interne du PC-1600, il est chargé dans la même zone de mémoire ou la même zone d'extension de mémoire qui était sélectionnée avant la mise hors tension.

9 Modes de fonctionnement en langage BASIC

Manipulation du clavier

Les touches, sur l'ordinateur, sont disposées de la même façon que pour une machine à écrire ou les autres ordinateurs. Il y a cependant certaines touches et combinaisons de touches qui sont particulièrement utiles dans le BASIC. De plus, certaines touches et leurs fonctions peuvent être contrôlées dans le cadre d'un programme BASIC.

Touches alphanumériques

Les touches alphanumériques sont disposées comme le clavier d'une machine à écrire normale et peuvent être réglées pour une répétition automatique, c'est-à-dire que le caractère est reproduit d'une manière itérative tant que la touche reste enfoncée. La fonction de répétition automatique est validée ou invalidée par la commande KEYSTAT. Cette commande permet également de régler toutes les touches du clavier de façon à émettre un cliquetis au moment de l'enfoncement, si cela s'avère nécessaire. Les touches numériques sont disposées comme le clavier numérique, à droite du clavier.

Touches de commutation

Les touches de commutation permettent aux touches normales de pratiquer plus d'une fonction. La plus courante d'entre elles est ta touche SHIFT qui se retrouve sur la plupart des machines à écrire pour sélectionner les majuscules ou les minuscules. La fonction de la touche SHIFT sur l'ordinateur est un peu différente. Le clavier de l'ordinateur possède huit touches de commutation.

MODE

La touche de mode permute l'ordinateur entre les modes RUN et PROgramme. Le mode RUN est le mode direct pour l'exécution des programmes et commandes en BASIC. Le mode PROgramme est le mode d'instruction dans lequel les programmes sont introduits, listés et édités. En outre, il y a un mode RESERVE qui est utilisé exclusivement pour les touches de fonction programmable.

SHIFT + MODE

L'ordinateur entre en mode de programmation RESERVE. Dans ce mode, les contenus des touches de fonction programmable peuvent être définis, modifiés et édités.

Appuyez uniquement sur la touche MODE pour sortir du mode RESERVE.

SHIFT

La touche SHIFT est un interrupteur à bascule; une fois pressée, elle reste effective et le mot SHIFT est affiché en haut sur l'écran jusqu'à ce qu'elle soit enfoncée encore une fois. La seconde pression rend effective la fonction écrite en orange au-dessus de la touche à la place de la fonction normale et l'affichage SHIFT disparaît. Pour utiliser séquentiellement la fonction supplémentaire de plusieurs touches, la touche SHIFT doit être enfoncée avant de manipuler chacune de ces touches.

CTRL

La touche de service CTRL est utilisée avec les touches ↑, ↓, →, ←, A, D, E, F, H, R et X pour les fonctions d'édition (voir le mode édition). Elle fonctionne comme un interrupteur à bascule, de même que la touche SHIFT. Une fois pressée la touche CTRL, le message CTRL apparaît en haut de l'écran et disparaît lorsqu'elle est enfoncée encore une fois.

(TOUCHE DE MENU)

La touche de menu fait périodiquement la commutation entre les trois modes RESERVE dans l'ordre I, Il et III en recommançant toujours par I. Chacun des trois modes de réserve permet de délivrer différent jeu de chaînes de menus des touches de fonction, procurant au total 18 chaînes possibles des touches de fonction.

SML

La touche (minuscules) SML fait la commutation entre les lettres majuscules et minuscules du clavier. En mode minuscules, le message SMALL est affiché en haut de l'écran.

DEF

La touche (définition) DEF a les deux fonctions suivantes; i) en mode RUN, elle permet de lancer le programme à partir d'une étiquette définie (voir le chapitre 8) et ii) en mode RESERVE, elle permet de définir une chaîne des touches de fonction. Lorsque la touche est pressée, le message DEF est affiché en haut de l'écran et persiste jusqu' à ce qu'elle soit appuyée encore une fois.

RCL

La touche (rappel) RCL a pour rôle de visualiser le menu des touches de fonction de F1 à F6 en mode RESERVE. L'affichage est effacé lorsqu'elle est pressée une seconde fois. L'affichage du menu des touches de fonction correspond à celui en mode RESERVE I, Il ou III selon ce qui est actuellement effectif.

CLICK
KBII

Cette touche a pour rôle d'activer et désactiver la fonction "cliquetis", lorsqu'elle est pressée après la touche SHIFT (SHIFT + CLICK). Utilisée toute seule, cette touche permet de sélectionner le clavier II (KBII) pour le jeu des caractères internationaux.

Touches spéciales

En bas du clavier et à droite des touches numériques, est disposée une variété de touches, dont certaines possèdent de fonctions spéciales dans le BASIC ou commandent le mouvement du curseur.

↓

La touche ↓ a les trois fonctions suivantes: i) en mode PROgramme, cette touche, lorsqu'elle est pressée sans aucune ligne de programme sur l'écran, permet d'afficher la première ligne du programme résidant alors en mémoire. Lorsqu'elle est enfoncée encore une fois, l'image de l'écran est déplacée jusqu'à la ligne subséquente du programme et ainsi de suite; ii) en mode PRO, si une erreur d'exécution est déclarée ou que le programme est interrompu par la commande BREAK, STOP ou INPUT, cette touche ↓ permet d'afficher la ligne dernièrement exécutée sur l'écran pour le but de l'édition; iii) en mode RUN, si une commande TRON (mode d'analyse) est spécifiée pendant l'exécution du programme ligne par ligne ou que l'exécution est interrompue par la commande BREAK ou STOP, la touche ↓ permet de reprendre l'exécution à partir de la ligne suivante.

↑

La touche ↑ a les quatre fonctions suivantes: i) en mode PRO, cette touche, lorsqu'elle est pressée sans aucune ligne sur l'écran, permet d'afficher la dernière ligne du programme résident alors en mémoire. Enfoncée deux fois, cette touche permet de déplacer l'image de l'écran et d'afficher la ligne précédant la dernière indiquée, en haut de l'écran; ii) en mode RUN, si l'exécution d'un programme est arrêtée par une erreur ou avec un message "BREAK IN <numéro de ligne>", la ligne, à laquelle l'exécution est arrêtée, est affichée tant que cette touche est maintenue enfoncée; iii) si une erreur est déclarée ou qu'un programme est interrompu par la commande BREAK, STOP ou INPUT, la touche ↑ possède la même fonction que la touche ↓; iv) en mode RUN, si une commande TRON (mode d'analyse) est spécifiée pendant l'exécution du programme ligne par ligne, la touche ^ permet d'afficher le contenu entier de la ligne en cours d'exécution.

Introduit le symbole π (pi) en MODE 1.
Possède un même effet que la touche ↓, en MODE 0.

SHIFT + ↓

Introduit le symbole V (racine carrée) en MODE 1.
Possède un même effet que la touche ↑, en MODE 0.

SHIFT + ↑

La touche **ENTER** est utilisée pour signaler la fin d'une entrée lorsqu'une séquence de caractères est introduite dans l'ordinateur. Les caractères ne sont pas transférés du tampon du clavier à l'ordinateur proprement dit, à moins que la touche **ENTER** ne soit pressée. En d'autres termes, l'ordinateur attend que la touche **ENTER** soit pressée, avant de commencer l'exécution de la commande ou des données spécifiées.

ENTER

Supprime le caractère immédiatement à gauche de la position du curseur et déplace le reste de cette ligne logique d'un caractère à gauche. Si le curseur est situé sur le premier caractère d'une ligne physique, la touche **BS** supprime le caractère ou l'espace à cette position et déplace le reste de la ligne d'un caractère à gauche. Pour une ligne logique, la touche **BS** déplace le curseur à gauche en supprimant les caractères jusqu'au début de cette ligne logique.

BS

Met l'ordinateur hors tension.

Met l'ordinateur sous tension. A la mise sous tension, la ligne de sommet apparaissant sur l'écran représente les modes qui ont été actifs lors de la dernière mise hors tension. Après la mise sous tension, la touche **ON** travaille comme la touche **BREAK**.

POWER OFF

POWER ON

Pressée une fois, la touche **BREAK** interrompt l'exécution du programme ou d'une commande en cours. Sur l'écran apparaît le message 'BREAK IN <numéro de ligne>'. La fonction de rupture **BREAK** peut être validée et invalidée par l'instruction BREAK ON/OFF.

BREAK

La touche **CL** supprime de l'écran, la ligne où le curseur est positionné, quand elle est pressée avant la touche **ENTER**. En mode RUN, si un programme est terminé avec une erreur, la touche **CL** efface le message de l'erreur sur l'écran.

La touche **CA** (**SHIFT** + **CL**) efface l'image de l'écran et remet les drapeaux internes de l'ordinateur à l'état "prêt à exécuter".

CL

CA

La touche ▶ a les deux fonctions suivantes: i) en mode PRO, lorsque le curseur se trouve sur une ligne de programme, cette touche permet de le déplacer d'une position à droite. Si le curseur est au bout de la ligne logique, cette touche n'a aucun effet; ii) en mode de calcul direct, cette touche permet de rappeler, sur l'écran, l'expression de calcul spécifiée avant sa pression S'il y a une erreur de syntaxe dans l'expression et que le message de l'erreur est affiché, il est possible de rappeler, avec cette touche, l'expression avec le curseur clignotant à la position de l'erreur.

▶

La touche ◀ a les deux fonctions suivantes: i) en mode PRO, lorsque le curseur se trouve sur une ligne de programme, cette touche permet de le déplacer d'une position à gauche. Si le curseur est au début de la ligne logique, cette touche n'a aucun effet; ii) en mode de calcul direct, cette touche possède un même effet que la touche ◀.

◀

La touche de suppression DEL (SHIFT + ◀) supprime le caractère de la position actuelle du curseur et serre l'espace correspondant.

La touche d'insertion INS (SHIFT + ▶) ménage un nouvel espace pour introduire un seul caractère immédiatement à gauche de la position du curseur et déplace le curseur à cet espace. La touche INS ne permet d'insérer qu'un seul caractère chaque fois qu'elle est pressée. Pour insérer une série de caractères, la touche INS doit être pressée avant de taper chaque caractère.

DEL

INS

Modes d'affichage

Cette section concerne les deux modes d'affichage du PC-1600 ainsi que la façon dont ils sont utilisés et, de même, explique les systèmes de coordonnées utilisés pour spécifier les emplacements des caractères et graphiques dans ces deux modes.

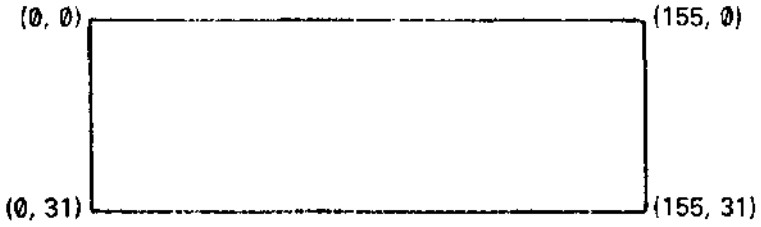
Les deux modes d'affichage sont référencés comme MODE 0 et MODE 1. L'utilisation de la commande MODE en rapport avec les modes d'affichage ne doit pas être confondue avec les modes de fonctionnement (RUN, PRO, RESERVE) de l'ordinateur. Pour une description totale de la méthode de spécification du mode d'affichage, se référer à l'article relatif à la commande MODE dans le répertoire des commandes. Le mode d'affichage par défaut est MODE 0 (le mode qui intervient après avoir réinitialisé l'ordinateur en spécifiant la commande ALL RESET).

MODE 0 (26x4)

Dans ce mode d'affichage, le nombre de lignes disponibles sur l'écran est fixé à 4 et le nombre de colonnes, à 26. Ceci procure, au maximum, 26 caractères ASCII par ligne. Chaque caractère est affiché en tant que matrice de points de 5 x 7.

Coordonnées de caractères en MODE 0

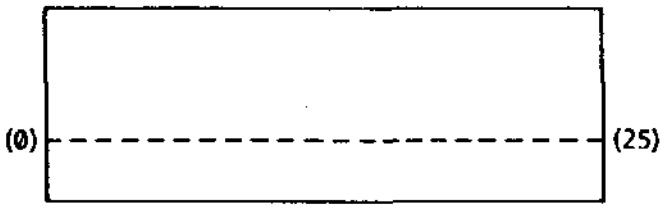
En mode graphique (voir les commandes GCURSOR, GPRINT, UNE, POINT, PRESET et PSET), il apparaît des données binaires de l'image sur l'écran, dans une matrice de 156 x 32, plutôt que les co'des caractères ASCII. Par exemple, avec l'instruction PSET, un point d'affichage peut être visualisé ou masqué à n'importe quelle position sur l'écran en spécifiant ses coordonnées X (allant de 0 à 155) et Y (allant de 0 à 31).



Coordonnées de graphiques en MODE 0 et MODE 1

MODE1 (26x1)

Dans ce mode d'affichage, le nombre de lignes sur l'écran est fixé à 1 et le nombre de colonnes, à 26. Ce mode a pour rôle fondamental de procurer la compatibilité avec l'affichage de l'ordinateur PC-1500 de Sharp. La ligne effective est limitée à la ligne de fond sur l'écran. Les autres trois lignes ne sont pas effectives pour les commandes PRINT et GPRINT, mais en mode PRO ou en cas d'utilisation de l'instruction INPUT, l'image sur l'écran est déplacée d'une ligne à la fois. Dans ce mode, le symbole pi (code ASCII &5D) et le symbole racine carrée (code ASCII &5B) sont affichés respectivement par "rc" et "√~" sur le PC-1500. Le jeu de caractères effectifs est aussi le même que pour le PC-1500. En MODE 1, il apparaît des données binaires de l'image sur l'écran, dans une matrice de 156 x 32, comme en MODE 0.



Coordonnées de caractères en MODE 1

Les instructions des graphiques dans le langage BASIC permettent à l'utilisateur de programmer une ample variété d'affichages personnalisés. Les instructions POINT et PSET permettent de visualiser les configurations réelles de points sur l'écran à partir d'un programme. Les commandes d'affichage graphique suivantes sont disponibles: GCURSOR, GPRINT, UNE, POINT, PRESET et PSET.

Mode d'édition

Une ligne d'instructions BASIC peut être d'une longueur de 80 caractères au maximum. Toutefois, étant donné que l'écran de l'ordinateur ne peut tenir que seulement 26 caractères par ligne, une ligne du BASIC peut occuper plus d'une ligne sur l'écran.

Pour distinguer une ligne en BASIC d'une ligne sur l'écran, les termes "ligne logique" et "ligne physique" sont utilisés. Une ligne logique est une ligne des instructions BASIC qui commence par un numéro de ligne et contient au maximum 80 caractères valables. D'autre part, une ligne physique est une ligne de 26 caractères au maximum apparaissant sur l'écran. Une ligne logique pourrait donc s'étendre sur quelques lignes physiques. Les fonctions d'édition en BASIC ne savent faire la distinction des lignes physiques.

Les lignes de programme ne peuvent être éditées qu'en mode PROgramme. Plusieurs méthodes sont disponibles pour l'édition des lignes d'un programme déjà introduit par le clavier.

La plus simple méthode est de retaper la ligne logique entière en utilisant un même numéro de ligne. Avec la commande LIST, vous pouvez facilement passer à une ligne et utiliser les fonctions d'édition pour positionner le curseur sur la ligne à laquelle vous désirez apporter des changements. L'ordinateur remplace automatiquement l'ancienne ligne par la nouvelle ligne, lorsque la touche **ENTER** est pressée. Faute d'appuyer sur la touche **ENTER**, les changements ne sont pas stockés et la ligne reste inchangée dans la mémoire.

Après l'exécution d'arrêt par un code d'erreur ou après la spécification de la commande BREAK, STOP ou INPUT, la touche **MODE**, une fois pressée, permute l'ordinateur en mode PRO. Ensuite, appuyez sur la touche ↓ ou ↑; alors, la dernière ligne est affichée avec le curseur placé à la position de rupture, signalant ainsi l'état prêt à l'édition.

Si, dans cette condition, vous appuyez sur la touche ENTER, les lignes de programme subséquentes apparaissent sur l'écran. Pour les lignes non affichées, vous pouvez les avoir sur l'écran en manipulant la touche ↑ ou ↓ de façon habituelle. Cet état est annulé en appuyant sur les touches SHIFT + CL.

La touche CTRL utilisée avec les touches BASIC normales énumérées ci-dessous, vous permet d'entrer et éditer les lignes d'un programme BASIC avec facilité, en passant directement aux lignes désirées d'un long programme sans avoir à progresser d'une ligne à l'autre à travers un long listage. La commande GOTO peut être utilisée avec une étiquette pour l'édition d'un seul des programmes résidant conjointement dans la mémoire. D'autre part, les commandes LIST, RENUM et DELETE vous permettent de manipuler les lignes sélectionnées d'un programme.

CTRL + ↑ Cette combinaison de touches permet d'effacer l'image en cours et d'afficher la première ligne du programme avec le curseur au début de cette ligne.

CTRL + ↓ Cette combinaison de touches permet d'effacer l'image en cours et d'afficher la dernière ligne du programme avec le curseur au début de cette ligne.

CTRL + ▶ Cette combinaison de touches permet de déplacer le curseur à la position suivant le dernier caractère de la ligne logique en cours de manipulation.

CTRL + ◀ Cette combinaison de touches permet de déplacer le curseur au début de la ligne logique sur laquelle il est alors placé. De même, cette combinaison supprime tous les caractères existant entre la position réelle du curseur et la fin de la ligne logique correspondante.

CTRL + A Cette combinaison de touches permet de commuter le mode entre INSERT et OVERWRITE pendant l'édition d'une ligne de programme. En mode INSERT, les caractères introduits par le clavier sont insérés dans la ligne, à la position juste avant le curseur et les caractères existants sont repoussés à droite. En mode OVERWRITE, le caractère introduit par le clavier est superposé sur le caractère de la position du curseur sans changer la longueur de la ligne correspondante. Lors de la mise sous tension de l'ordinateur, celui-ci est automatiquement mis en mode OVERWRITE.

Le curseur clignote par intervalle plus réduit lorsque le mode INSERT est choisi.

Cette combinaison de touches supprime tous les caractères existant entre la position réelle du curseur et la tête de la ligne correspondante, y compris le numéro de ligne.

CTRL + D

Cette combinaison de touches annule tous les caractères à partir de la position actuelle du curseur jusqu'à la fin de la ligne logique.

CTRL + E

Cette combinaison de touches permet de déplacer le curseur à droite jusqu'au premier caractère du mot suivant sur la même ligne.

CTRL + F

Cette combinaison de touches supprime le caractère immédiatement à gauche du curseur et déplace le reste de la ligne logique d'une position de caractère à gauche. Si les touches sont pressées avec le curseur positionné au premier caractère d'une ligne logique, le caractère ou l'espace de cette position est supprimé et le reste de la ligne est déplacé d'un caractère à gauche. Pour une ligne logique, ces touches permettent de déplacer le curseur à gauche le long de cette ligne jusqu'à son début, tout en effaçant les caractères et, si nécessaire, en passant par dessus les lignes physiques. La touche de rappel arrière BS fonctionne de la même façon.

CTRL + H (BS)

Cette combinaison des touches met en circuit et hors circuit la fonction de répétition de touche, de même que la commande KEYSTAT.

CTRL + R

Cette combinaison des touches supprime tous les caractères de la ligne logique sur laquelle le curseur est positionné. La touche CL possède le même effet.

CTRL + X

Mode réserve

En mode RESERVE, l'ordinateur permet l'accès à une zone spéciale de la mémoire qui stocke les chaînes de caractères affectées aux touches de fonction situées en haut du clavier, juste au-dessous de l'écran et marquées par les étiquettes !, ", #, $, %, &.

La chaîne de caractères spécifiée par l'une des touches de fonction peut être changée ou modifiée par l'utilisateur. Les chaînes peuvent aussi être stockées sur un périphérique de la même façon que pour un programme normal et chargées à tout moment dans l'ordinateur pour programmer les touches de fonction pour une certaine application particulière.

Sélection du mode RESERVE

Appuyez sur les touches SHIFT + MODE. A la ligne d'état, au sommet de l'écran, est affiché le message RESERVE. Vous pouvez sortir du mode RESERVE en pressant encore une fois sur la touche MODE.

Lorsque le message RESERVE est affiché sur la ligne d'état, il est possible de choisir l'un des trois modes RESERVE, soit le mode RESERVE I, le mode RESERVE II ou le mode RESERVE III. Le mode I, II ou III est sélectionné par la touche de menu ⇔ et indiqué sur la ligne d'état adjacente au message RESERVE. La touche de menu permute périodiquement les trois modes RESERVE dans l'ordre I, II et III en recommençant par I. Chacun des trois modes RESERVE permet de sortir un différent jeu de chaînes de caractères affectées aux touches de fonction, d'où la possibilité de spécifier au total 18 chaînes de caractères.

Programmation des chaînes de caractères sur les touches de fonction.

1Sélectionnez d'abord le mode RESERVE en appuyant sur les touches+ SHIFT MODE. Tapez NEW à partir du clavier et appuyez sur la touche ENTER. Ceci efface toutes les chaînes existant dans la mémoire RESERVE pour les trois modes. Si vous désirez conserver les contenus des touches de fonction ou modifier seulement certaines touches, sautez cette étape.

2Sélectionnez le mode I pour définir les contenus des touches de fonction de F1 à F6.

3 Appuyez sur la touche F1 et tapez la chaîne de caractères que vous voulez faire apparaître sur l'écran en pressant F1 pendant les opérations normales. Tous les caractères alphabétiques et numériques, de même que les symboles, peuvent être utilisés. Appuyez ensuite sur la touche ENTER.

4 Répétez les opérations de l'étape 2 afin de programmer les chaînes de caractères pour les touches de fonction de F2 à F6.

5Sélectionnez le mode II en appuyant encore une fois sur la touche de menu ⇔.

6Répétez encore six programmations pour les six touches de fonction en mode II (étapes 3 et 4).

7Sélectionnez le mode III avec la touche de menu et répétez encore six programmations.

Maintenant, vous devez avoir affecté 18 chaînes de caractères aux touches de fonction dans les modes I, II et III.

La programmation possible pour les touches peut comprendre les mots des commandes BASIC et les noms de fonction ou même les expressions de calcul réel.

Exemple de programmation des touches de fonction:

Touche	MODE I	MODE II	MODE III
F1	SIN	FOR	PRINT
F2	COS	TO	LIST
F3	TAN	STEP	AUTO
F4	LOG	NEXT	LOAD
F5	EXP	GOTO	SAVE
F6	INT	RUN	LLIST

La longueur maximale de la chaîne de caractères affectée à une touche de fonction est de 110 octets. Une commande ou une fonction BASIC (par exemple, RUN, PRINT, SIN, COS) occupe 2 octets. Un numéro, une lettre ou un symbole occupe 1 octet. Par conséquent, une touche de fonction peut stocker une phrase plus longue qu'une simple commande. Par exemple, F1 pourrait être programmé de façon à retenir l'instruction suivante en BASIC:

F1 FOR I = 1 TO 100STEP5

Vous pouvez réduire considérablement le temps de programmation en choisissant les commandes et instructions que vous utilisez le plus fréquemment au cours de la programmation.

Rappel des chaînes des caractères affectées aux touches de fonction

La chaîne de caractères de la touche de fonction stockée dans la mémoire RESERVE pour le mode I, Il ou III peut être rappelée en mode RUN ou PRO pendant l'introduction directe ou l'écriture du programme. Sélectionnez le mode nécessaire par la touche ⇆ et appuyez sur la touche de fonction appropriée. Les contenus sont alors affichés sur l'écran, à la position du curseur. Si, dans l'exemple donné ci-dessus, la touche F1 est pressée avec le mode III spécifié, le mot PRINT apparaîtra sur l'écran.

La zone de la mémoire utilisée pour le stockage des chaînes de caractères des touches de fonction en mode II, est aussi utilisée pour retenir le message optionnel pour une commande ALARM$. Si une alarme est définie avec un message, celui-ci est affiché et écrit dans la mémoire à la place de la chaîne de caractères affectée à la touche de fonction.

Menus des touches de fonction

Etant donné que les contenus programmés des touches de fonction ne sont pas affichés avant qu'elles ne soient pressées, le PC-1600 dispose d'une fonction permettant de faire apparaître sur l'écran le menu à simple ligne pour chaque touche de fonction en modes I, II et III.
Le menu peut être affiché en appuyant sur la touche de rappel **RCL**. Ce menu peut, à titre d'exemple, indiquer le contenu de chaque touche sous une forme abrégée. Le menu est stocké dans l'ordinateur en une chaîne de 26 caractères et son seul rôle est de signaler à l'utilisateur quel affichage donne chaque touche de fonction lorsqu'elle est pressée.

Création d'un menu:

1 Placez d'abord l'ordinateur en mode RESERVE en appuyant sur les touches **SHIFT** + **MODE**.

2 Sélectionnez le mode pour lequel vous désirez introduire le menu par la touche de menu

3 Tapez la chaîne de menu dans une limite de 26 caractères au maximum, en la mettant entre guillemets ("), et appuyez sur la touche **ENTER**.

Le menu est stocké dans une partie de la mémoire de réserve pour le rappel ultérieur. Il est à rappeler que la longueur maximum de la chaîne de menu est de 26 caractères (correspondant à la largeur de l'écran). De plus, aucun article du menu ne doit excéder 4 caractères, si vous désirez affecter correctement l'article à la touche de fonction correspondante.

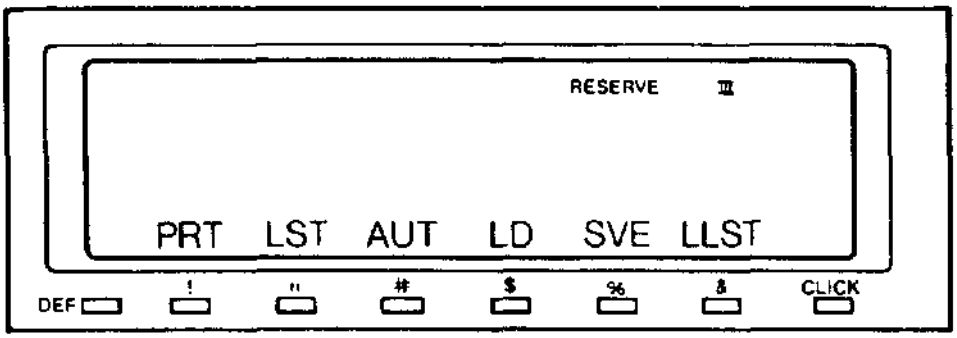
Dans l'exemple précédemment donné, le menu des chaînes de caractères affectées aux touches de fonction en mode III peut apparaître comme suit:

```
PRT LST AUT LD SVE LLST
←      26 caractères      →
```

Affichage du menu:

Le menu donné à titre d'exemple peut être affiché en mode RUN ou PRO en sélectionnant le mode III par la touche de menu et appuyant sur la touche **RCL**.Si vous appuyez sur la touche **RCL** une seconde fois, le menu est effacé. De cette sorte, les menus pour les modes I et II peuvent être affichés pour rappeler à l'utilisateur les chaînes de caractères attribuées aux touches de fonction.

Le menu d'exemple pour le mode III apparaît sur l'écran de PC-1600, aux positions correspondant aux touches de fonction comme suit:



Sauvegarde et chargement des programmes de touches de fonction:

La chaîne de caractères attribuée à chaque touche de fonction peut être sauvegardée et chargée sur cassette de la même façon que pour un programme ordinaire en BASIC. D'abord, mettez l'ordinateur dans le mode RESERVE en appuyant sur les touches **SHIFT** + **MODE**, puis spécifiez la commande CSAVE ou
CLOAD comme en cas de sauvegarde ou chargement d'un programme BASIC. Il n'est pas possible d'utiliser un disque ou la mémoire RAM pour stocker les spécifications des touches de fonction.

REMARQUE: Pour le chargement d'un programme de la touche de fonction, il est important de vérifier le mode de l'ordinateur qui doit être le mode RESERVE; si en mode PRO, le programme est chargé dans la zone réservée au programme normal BASIC, détruisant ainsi tous les programmes BASIC déjà existants.

Mots clés préaffectés

Quelques mots le plus souvent utilisés dans les instructions BASIC sont attribués en permanence à chacune des touches alphabétiques dans la rangée la plus haute du clavier (QWERTYUIOP). Ces mots clés peuvent être extraits en n'importe quel mode en appuyant sur la touche **DEF** suivie de la touche alphabétique correspondante. Les mots clés disponibles et les touches alphabétiques correspondantes sont:

Touche	Mot clé
Q	1NPUT
W	PRINT
E	USING
R	GOTO
T	GOSUB
Y	RETURN
U	CSAVE
1	CLOAD
0	MERGE
p	LIST

Les mots clés CSAVE, CLOAD et MERGE peuvent être utilisés seulement lorsqu'un enregistreur à cassette est connecté avec l'unité interface de cassette par l'intermédiaire de l'imprimante CE-1600P.

Ces mots clés sont mis en relief sur la grille fournie avec votre ordinateur. Placez-la sur les touches correspondantes du clavier pour faciliter les opérations.

De plus, la touche **DEF** suivie d'une touche de fonction permet d'extraire encore six mots clés suivants:

DEF + **!** → RUN **ENTER**
DEF + **"** → AUTO
DEF + **#** → LOAD"
DEF + **$** → SAVE"
DEF + **%** → FILES"
DEF + **&** → "COM1:"

10 Représentation des données

Les données à l'intérieur de l'ordinateur sont traitées en unités de nuits chiffres binaires. Une unité de huit bits est appelée un "octet". Dans le système de numération binaire, il est possible de représenter tous les nombres, allant de 0 à 225, par huit bits. Par exemple, le nombre 0 est exprimé par 00000000, 17 par 00010001 et 255, par 11111111.

Normalement, pour simplifier l'écriture de ces chaînes de huit bits dans les tables et graphiques, elles sont converties en valeurs hexadécimales équivalentes. La représentation hexadécimale est toujours plus courte que la représentation binaire d'un même octet. Pour la distinction entre les systèmes décimal et hexadécimal, un "&" est placé avant le nombre hexadécimal. Par conséquent, &27 signifie 27 dans le système de numération hexadécimale.

Types de données

Ensemble de données texte et jeux de caractères

Afin d'utiliser ce système à huit bits pour représenter les caractères alphabétiques et les symboles, tels le signe de ponctuation et les parenthèses, un code standard est utilisé avec chaque caractère représenté par un nombre binaire de huit bits.

Il existe actuellement trois codes principaux standard, à savoir le code ASCII (USA), le code EBCDIC (Europe) et le code J!S (Japon)/En ce qui concerne les principaux caractères alphabétiques et les autres caractères d'usage commun, ces trois codes standard sont totalement interchangeables. Il y a cependant une certaine différence, entre ces trois codes, dans les symboles graphiques et spéciaux (le symbole de devise par exemple). Le PC-1600 utilise un jeu de codes caractères similaire à celui des ordinateurs individuels IBM.

Le jeu de caractères utilisé dans le PC-1600 est un peu différent de celui pour le PC-1500. Ceci signifie que certains caractères d'affichage du PC-1600 varient entre le MODE 0 et le MODE 1 (voir les modes d'affichage). Les différences se situent dans les caractères représentés par &5B (symbole de la racine carrée), &5D (symbole Pi) et certaines sortes de parenthèses. Voyez les tableaux dans l'Annexe C.

Chaînes de caractères

La chaîne de 80 caractères au maximum peut être traitée en tant qu'ensemble unique par l'ordinateur. Sa façon de traitement est différente de celle pour les données numériques et les calculs numériques ne peuvent pas être exécutés avec

les chaînes de caractères, alors celles-ci acceptent une certaine manipulation; de manière concise, elles peuvent être ajoutées de façon à former de plus longues chaînes ou partiellement extraites.

Données numériques

Toutes les données numériques sont converties en forme binaire pour le stockage ou le calcul dans l'ordinateur. De plus, l'ordinateur accepte l'entrée numérique soit en notation numérique normale, soit en notation en virgule flottante. Dans certains cas particuliers, l'entrée et la sortie peuvent être spécifiées sous forme hexadécimale (voir la description des commandes HEX$).

L'ordinateur utilise la représentation en virgule flotante pour les calculs numériques; un nombre est intérieurement représenté sous une forme exponentielle par 12 chiffres multipliés par la puissance 10. Cependant, la valeur est arrondie à 10 chiffres pour la sortie. C'est ainsi que l'ordinateur entre et sort les nombres allant de ± 9,999999999 x 10"" à ± 9,999999999 x 10"". Cet arrondissement donne une erreur opérationnelle de ± 1 au dixième chiffre (ou strictement, ± 1 dans la mantisse).

Constantes

Constantes alphanumériques

Une constante alphanumérique peut être une suite quelconque de caractères alphanumériques, délimitée par les guillemets. Les exemples suivants représentent quelques constantes alphanumériques qui sont toutes valables:

"SHARP PC-1600"
"ABCDEF"
"Ordinateur de poche révolutionnaire"
"1.256 à 84.4"

L'ordinateur ne peut pas distinguer les caractères numériques de ceux alphabétiques, à moins qu'ils ne soient délimités par les guillemets. Une paire de guillemets spécifiée en séquence, à savoir " ", est considérée comme une chaîne nulle de longueur zéro. Ceci a cependant certaines applications dans la programmation (voir INKEY$).

La longueur maximale d'une constante alphanumérique est de 30 caractères.

Constantes numériques

Les constantes numériques peuvent être un nombre positif, ou négatif. Quatre types de constantes numériques sont disponibles:

1 Constantes entières

Les nombres disponibles en tant que constante entière vont de -32768 à +32767. Ils peuvent être précédés soit par un signe +, soit par -. S'il y a aucun signe, le nombre donné est considéré comme étant positif.

Exemple: 123, +2, -57, 1024

2 Constantes à virgule fixe

Il y a des nombres positifs ou négatifs comprenant une fraction décimale. Ils peuvent être précédés soit par un signe +, soit par -. A défaut de signe, le nombre donné est considéré comme étant positif.

Exemple: 12.7, +1.345,0.222232, -67.9888

3 Constantes à virgule flottante

Il y a des nombres positifs ou négatifs représentés sous forme exponentielle. Ils sont composés d'une constante entière ou à virgule fixe suivie de la lettre E et d'un exposant. La mantisse ou l'exposant peut être positif ou négatif. A défaut du signe, la partie concernée est considérée comme étant positive.

Exemple: 2.43E9, +1.99E-3, +50.23E+2

4 Constantes hexadécimales

Il y a des constantes représentées selon le système de numération hexadécimale. Pour distinguer ces constantes des constantes entières décimales, elles sont précédées par le signe "&".

Exemple: &23, &8000, &2B, &FFFF

Les constantes hexadécimales peuvent se trouver sur une fourchette de &0 (zéro en décimale) à &FFFF (65535 en décimale). Les nombres négatifs ou fractionnels ne sont pas valables.

Variables

Les variables sont des zones définies dans la mémoire de l'ordinateur, auxquelles des noms sont attribués par l'utilisateur pour retenir les valeurs pendant l'exécution d'un programme. Les valeurs retenues en variables peuvent être spécifiées par l'utilisateur ou écrites automatiquement par le programme lui-même.

Deux types de variables sont utilisés dans le langage BASIC de Sharp, à savoir la variable numérique et la varibale alphanumérique. La variable numérique est utilisée pour stocker les nombres à utiliser dans les calculs ou les résultats de calculs. Elle est traitée sous forme virgule flottante dans l'ordinateur et peut être sortie sous forme virgule fixe ou flottante selon le type de calcul exécuté. Le BASIC convertit automatiquement d'un type à l'autre selon la nécessité. Les variables numériques peuvent être introduites dans l'ordinateur soit sous forme virgule fixe, soit sous forme virgule floattante.

Exemple: A = 12.6043, VA = 1.2345436687E-3

La variable alphanumérique peut être utilisée pour le stockage des chaînes de caractères ASCII valables.

En plus d'affecter un nom à chacune des variables, il faut attribuer à certaines variables une partie de la mémoire dont la grandeur dépend de la taille des données devant être stockées. Ceci est exécuté par l'instruction BASIC, DIM. Une variable numérique reste zéro, à moins qu'une certaine valeur n'y soit stockée; il en est de même pour une variable alphanumérique.

Nom de variable

Le nom d'une variable numérique est composé de 2 caractères au maximum. Le premier caractère peut être l'une quelconque des lettres majuscules, allant de A à Z. Le seconde caractère peut être soit l'une des lettres majuscules de A à Z, soit l'un des chiffres de 0 à 9.

Le nom de variable peut aussi contenir plus de deux caractères; cependant, les caractères situés à droite des deux premiers caractères sont ignorés au niveau de l'identification de la variable. Par exemple, les noms TOTAL et TOP sont considérés comme un même nom par le BASIC. Par contre, A et AB sont différentes variables.

Exemple: A, AB, C1, D9, @G, RE(6), X2(I)

Le nom de variable alphanumérique est assujetti aux mêmes règles fixées pour les variables numériques et, en plus, doit avoir le symbole "$" pour le dernier caractère.

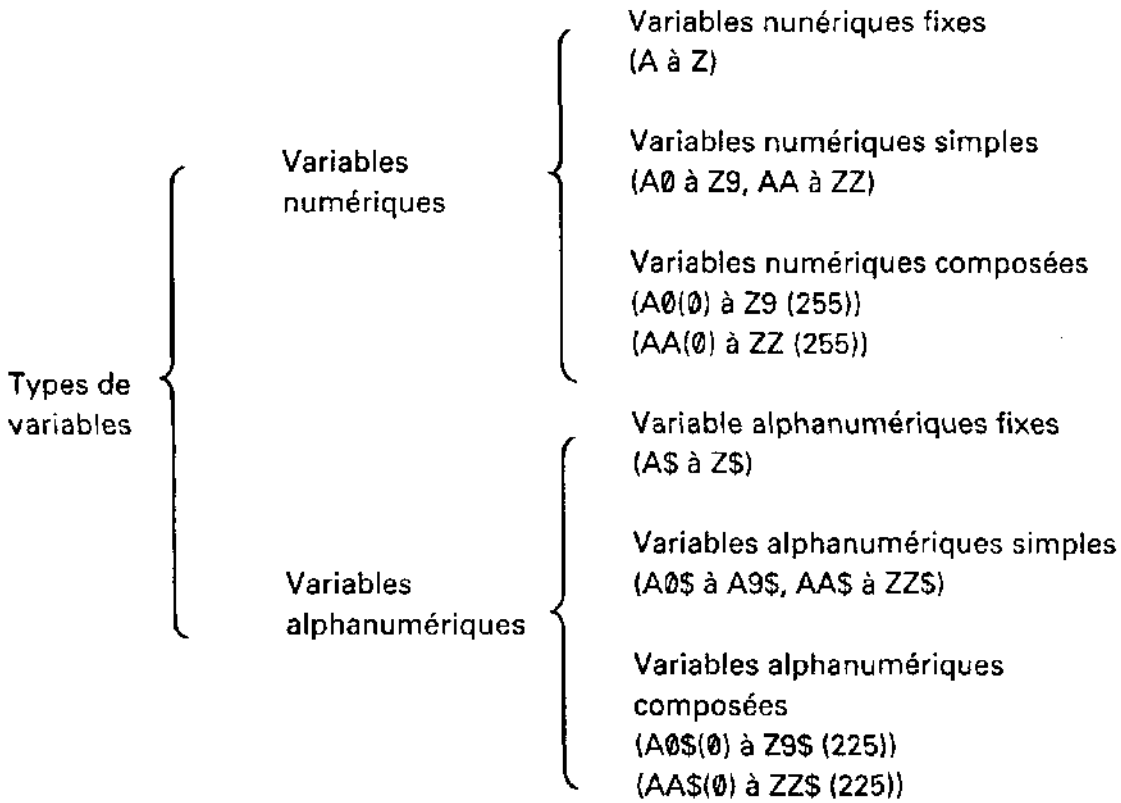
Exemple: CC$, B$, D4$, NO$(4), TR$(12)

Types de variables

Dans le BASIC de Sharp, les variables sont classées en trois types pour l'affectation de la mémoire. Les trois types de variables sont:

i) variables fixes ii)
variables simples iii)
variables composées

Certaines lettres sont réservées pour les types de variables spécifiques.
Le schéma donné ci-dessous représente ces trois types de variables avec leurs noms affectés dans le BASIC de Sharp.



1 Variables numériques fixes

Les emplacements de stockage des variables numériques et alphanumériques fixes sont déterminés sur la zone particulièrement affectée dans la mémoire (voir l'annexe D, Image mémoire).

Ces emplacements sont toujours disponibles pour stocker le nombre maximum de variables fixes qui peuvent être identifiées par le nom. La zone pour les variables fixes peut être utilisée en tant que tableau à une dimension avec le symbole @ pour le nom et un indice de 1 à 26. Ainsi, @(1) est stocké dans l'emplacement A et @(26), dans l'emplacement Z.

Les variables fixes peuvent être effacées de la mémoire par l'instruction CLEAR.
Les variables simples et composées peuvent également être effacées en même temps. Pour effacer seulement les variables simples et composées, mais non les variables fixes, utilisez l'instruction ERASE.

2 Variables numériques simples

Les variables numériques simples sont stockées dans une partie de la zone réservée au programme BASIC et la taille de cette partie affectée varie en fonction du nombre de variables utilisées.

La mémoire pour le programme de vient d'autant plus limitée que les variables sont utilisées; au contraire, lorsque le programme est plus encombrant, la zone disponible pour le stockage des variables simples est réduite d'autant. Lorsqu'il ne reste qu'un espace insuffisant pour stocker les variables simples produites par le programme du fait de sa staille très importante, élargissez la zone mémoire de l'ordinateur en ajoutant un module RAM (référez-vous au chapitre 6).

Le premier caractère du nom de variable simple doit être une lettre majuscule.

3 Variables numériques composées

Le tableau (ou matrice) est un ensemble de variables référenciées par un nom commun. Chaque membre du tableau est référencé par un indice spécifié en nombre entier juste après le nom de tableau.

Exemple: A(2) est le second membre du tableau A(X).
A(15) est le quinzième membre.

Le tableau peut être soit à une dimension (telle qu'indiquée dans l'exemple ci-dessus), soit à deux dimensions dans lesquelles deux indices sont requis pour spécifier un membre particulier du tableau correspondant.

Exemple: X(3,4), AB(4,12)

Un même nom ne peut pas être utilisé pour les tableaux à une dimension et à deux dimensions à la fois.

Le tableau à une dimension est une liste et le tableau à deux dimensions, une matrice. Etant donné que le tableau est une liste ou une matrice de membres, le

nombre total de membres dans un certain tableau spécifique doit absolument être connu afin de déterminer la grandeur de l'espace à réserver pour retenir tous les membres. L'instruction DIM est spécifiée toutes les fois que les tableaux sont utilisés dans un programme pour définir ou dimensionner le nombre maximum de membres envisagés.

Voici tous les membres du tableau A(X,Y):

	Colonne 0	Colonne 1	Colonne 2	Colonne 3 ...	Colonne Y
Rangée 0	A(0,0)	A(0,1)	A(0,2)	A(0,3)	A(0,Y)
Rangée 1	A(1,0)	A(1,1)	A(1,2)	A(1,3)	A(1,Y)
Rangée 2	A(2,0)	A(2,1)	A(2,2)	A(2,3)	A(2,Y)
Rangée 3					A(3,Y)
	A(3,0)	A(3,1)	A(3,2)	A(3,3)	
Rangée X					A(X,Y)
	A(X,0)	A(X,1)	A(X,2)	A(X,3)	

Il est à noter que dans l'ordinateur, la première rangée est la rangée 0 et la première colonne, la colonne 0. Par conséquent, le tableau dont le membre le plus grand est A(3,3) est un tableau à deux dimensions (une matrice en fait) avec 16
membres, dont chacun a son propre nom de tableau. L'espace doit donc être réservé dans la mémoire pour chacun des 16 membres possibles. Référez-vous à l'instruction DIM dans le Dictionnaire des commandes, pour une description détaillée.

Les noms de tableaux peuvent être attribués de la même façon que pour les noms de variables simples et, en plus, doivent être suivis d'un ou plusieurs indices, lesquels peuvent être eux-mêmes une variable. Ceci permet d'utiliser un programme simple de bouclage pour traiter automatiquement tous les membres d'un tableau dans l'ordre (voir l'exemple de programmation, pour les instructions READ...DATA dans le Dictionnaire des commandes).

A(X,Y), KA(J), B1(J,K) sont tous les variables composées qui peuvent représenter un membre quelconque dans le tableau. A(1,3), KA(6), B1(5,7) sont tous les membres particuliers de ce tableau.

De même que pour les variables numériques simples, les nombres stockés en tant que membre d'un tableau numérique peuvent être soit sous forme virgule flottante, soit sous forme virgule fixe.

Le nombre maximum de membres dans un tableau est de 65535. Pour un tableau à deux dimensions, il est fixé à 255 colonnes x 255 rangées.

L'espace de la mémoire réservé aux membres d'un tableau peut être remis à zéro par l'instruction ERASE qui, cependant, maintient toutes les variables fixes telles quelles. La taille maximum pour un tableau quelconque est de 64K octets.

4 Variables alphanumériques fixes

Les noms de variables alphanumériques fixes sont assujettis aux mêmes règles prévues pour les variables numériques fixes et, en plus, doivent avoir le symbole $ pour le second caractère.

Exemple: A$, C$ P$, @$(2)

Une variable alphanumérique fixe peut être d'une longueur de 16 caractères au maximum.

5 Variables alphanumériques simples

Les noms de variables alphanumériques simples sont assujettis aux mêmes règles que les noms de variables numériques simples et, en plus, doivent avoir le symbole $ pour dernier caractère.

Exemple: AB$, C7$, Z9$, XX$

Une variable alphanumérique simple peut être d'une longueur de 16 caractères aux maximum.

6 Variables alphanumériques composées

Les noms des variables alphanumériques composées répondent aux mêmes règles que les variables numériques composées mais, en plus, sloivent avoir le symbole $ pour dernier caractère.

Exemple: A1$(3), KS$(5,5), C9$(J), JH$(P,Q)

De plus, au moment de spécifier le nombre de membres dans un tableau de caractères à l'aide de l'instruction DIM, la longueur maximum de chaque membre du tableau doit aussi être spécifiée pour réserver un espace suffisant dans la mémoire. Faute de spécification de la longueur, l'espace réservé alors à chaque membre est pour 16 caractères. Le tableau à multi-membres peut parfois conduire à réserver une majeure partie de la mémoire et n'est jamais utilisé. Ceci est le cas, lorsque les membres d'un tableau de caractère sont d'une longueur de plus de 3 caractères. S'il existe 25 membres dans le tableau (par exemple, AB(4,4)) et que chaque membre a une longueur de 3 caractères au maximum, la négligence de spécification de la longueur de chaîne alphanumérique a pour résultat de réserver la mémoire non utilisée pour retenir 325 espaces de caractères non utilisés dont la décomposition est de 16 - 3 = 13 caractères pour chaque membre x 25 membres

Le nombre maximum de membres dans un tableau de caractères est de 65535. Pour un tableau à deux dimensions, il est fixé à 255 colonnes x 255 rangées.

L'espace de mémoire réservé aux membres d'un tableau peut être remis à zéro par l'instruction ERASE.

Expressions et opérateurs

Dans un programme, toute combinaison de variables, constantes numériques et constantes alphanumériques est appelée "expression". Une expression peut aussi contenir seules les variables ou constantes.

Les opérateurs sont des symboles servant à indiquer les opérations, mathématiques ou logiques, devant être exécutées dans l'expression ou dans un.e partie de l'expression. Les opérations exécutées dans BASIC sont divisées en quatre catégories fondamentales: (1) opérations arthmétiques, (2) opérations de relation, (3) opérations logiques et (4) opérations fonctionnelles.

Opérations arithmétiques

Les opérations sont exécutées dans l'ordre prioritaire fixé par le BASIC lui-même Dans une simple expression arithmétique^ l'ordre prioritaire est le suivant:

Opérateur	Opérations exécutées	Exemple
	Elévation à une puissance	X * 3
-	Négation (signe de changement)	-X
*,/	Multiplication, division	X*2, X/Y
MOD	Arithmétique de module	A MOD B
V	Division en nombre entier	A\ B
+,-	Addition, soustraction	X+A, Y-B

Les opérations arithmétiques énumérées ci-dessus priment sur les opérations de relation ou logiques.

L'ordre dans lequel sont évaluées les expressions arithmétiques peut être changé en mettant certaines parties de l'expression entre parenthèses ( ), en respectant les règles normales de l'algèbre. Dans ce cas, les parties mises entre parenthèses sont évaluées avant les autres. Les expressions algébriques ordinaires sont transformées en celles équivalentes de BASIC comme suit:

Expression algébrique ordinaire	Expression en BASIC
x^2	x"2
$(x + 3)^4$	(x + 3) " 4
$2(x + y)$	2 * (x + y)
bX^2	b"x"2

REMARQUE: • Les opérations exponentielles consécutives dans une même expression sont exécutées de droite à gauche:

Exemple: 3 " 4 " 2 -> 3^{42}

Lorsqu'une puissance et une négation apparaissent dans une même expression, l'évaluation est exécutée de droite à gauche:

Exemple: $-2\hat{}4 \rightarrow -(2^4)$

$3\hat{}-2 \rightarrow 3^{-2}$

Opérations de relation

Les opérateurs de relation permettent de comparer deux valeurs. Le résultat de comparaison est représenté soit par "vrai" (1), soit par "faux" (0). Ce résultat peut être utilisé dans un programme pour prendre une décision au cours de son déroulement subséquent, en utilisant les instructions IF..THEN..ELSE et IF..GOTO.

Les opérateurs de relation suivants peuvent être utilisés:

Opérateur	Opérations exécutées	Exemple
=	Egalité	A = B X<> Y
<>	Inégalité	A< B B> A Y<
<	Inférieur à	= 3 X> = 1
>	Supérieur à	
<=	Egal ou inférieur à	
>=	Egal ou supérieur à	

REMARQUE:
- Le signe "=" est utilisé aussi bien dans les expressions de comparaison que dans les expressions d'affectation avec l'instruction LET. Toutefois, la signification n'est pas la même dans ces deux cas.

Les comparaisons peuvent être exécutées entre deux expressions ou variables numériques (soit, X et Y) ou encore entre deux expressions ou variables alphanumériques (soit, A$ et B$). Toutefois, les expressions numériques ne peuvent pas être comparées avec les expressions alphanumériques.

Deux chaînes de caractères sont comparées, caractère par caractère, de gauche à droite, sur la base de leurs valeurs de code caractère (voir l'annexe C). Si deux chaînes comportent les mêmes caractères et que chacune des chaînes est d'une même longueur, elles sont égales.

Pour les chaînes d'une même longueur qui présentent cependant une discordance dans une paire de caractères, le code caractèrs de la première paire de caractères non conformes est comparé et la chaîne contenant le caractère de discordance dont la valeur de

code est la plus grande, est considérée comme étant plus importante. Si une courte chaîne est comparée avec une longue chaîne et que tous les caractères de la courte chaîne sont conformes à ceux testés dans la longue chaîne, cette dernière est considérée comme étant plus importante.

Un espace ménagé dans la chaîne de caractères est assimilé à un caractère sur la base de sa valeur de code caractère. La chaîne la plus courte est la chaîne "nulle" avec sa valeur de code caractère égale à 0.

- Les opérateurs arithmétiques priment sur les comparaisons dans l'ordre d'évaluation.

Opérations logiques

Les opérations logiques utilisent les fonctions de l'algèbre de Boole, AND, OR et NOT pour établir les connexions entre les expressions de relation. Les opérations logiques dans une simple expression sont évaluées suite aux opérations arithmétiques et de relation.

De cette façon, les opérateurs logiques peuvent être utilisés pour prendre les décisions au niveau du programme sur la base de différentes conditions, à l'aide des instructions IF..THEN..ELSE et IF..GOTO.

Exemple

IF A < = 32 AND B > = 90 THEN 150

Cette instruction permet de sauter jusqu'à la ligne 150 lorsque la valeur de la variable numérique A est égale ou inférieure à 32 et que la valeur de la variable numérique B est égale ou supérieure à 90.

IFX<> 13 OR Y = 0THEN 50

Cette instruction permet de sauter jusqu'à la ligne 50 à moins que la variable X ne soit égale à 13 ou lorsque la variable Y est égale à 0.

Dans une opération logique impliquant deux nombres de -32768 à 32767, ces deux nombres sont convertis en nombres binaires entiers à 16 bits (forme de complément à deux) et la connexion logique est ensuite évaluée pour chaque paire correspondante de bits dans les deux nombres.

Les résultats renvoyés par les opérateurs logiques pour ces évaluations de bits sont tels qu'indiqués ci-dessous:

AND

X	Y	X AND Y
1	1	1
1	0	0
0	1	0
0	0	0

OR

X	Y	X OR Y
1	1	1
1	0	1
0	1	1
0	0	0

NOT

X	NOT X
1	0
0	1

Après que chaque paire de bits ait renvoyé le résultat correspondant (soit 1 ou 0) selon les tables données ci-dessus, le nombre binaire à 16 bits résultant est converti en une valeur décimale, laquelle est le résultat de l'opération logique.

Exemple:

41 AND
égal
9

41 = 101001
27 = 011011
^-001001
AND

41 OR 27
-
égal
59

41 - 101001
27 = 011011
<-111011
OR

NOT 3
ega
-4 (forme de complément à deux)

3 = 0000000000000011
1111111111111100
NOT

NOT X peut généralement être calculé par l'équation, NOT X = -(X + 1)

Opérations fonctionnelles

Les fonctions sont des opérations consistant à calculer une valeur spécifique à partir d'un simple opérande intégré dans les programmes BASIC. Les fonctions sont généralement décrites ensemble avec les instructions BASIC dans le Dictionnaire des commandes. Les fonctions suivantes sont intégrées dans le BASIC de Sharp:

ABS	Valeur absolue	DMS	Degré/décimal
ACS	Arc cosinus	EXP	Exposant
ASN	Arc sinus	INT	Nombre entier
ATN	Arc tangentiel	LN	Logarithme naturel
COS	Cosinus	LOG	Logarithme commun
DEG	Décimal/degré	PI	Valeur de pi

RND	Nombre aléatoire	SQR	Racine carrée
SGN	Signe	TAN	Tangente
SIN	Sinus		

Les arguments peuvent être spécifiés ou renvoyés en degrés, radiants ou valeur de gradients en sélectionnant le mode avec les commandes DEGREE, RADIAN et GRAD.

Dans les calculs impliquant des fonctions et des opérations arithmétiques, toutes deux en programmes BASIC et en mode calculatrice, la priorité d'exécution est la suivante :

L'exponentiation a priorité sur les fonctions. Par conséquent, dans certaines expressions des parenthèses doivent être utilisées pour garantir une évaluation correcte.

Exemple:

Forme algébrique normale	'	Expression PC-1600
$\sin^2 30°$		(SIN 30)^ 2
$(\sin 30°)^2$		SIN 30 * 2
$\cos^4(A+B)$		(COS(A+B)) *4

Outre les fonctions mathématiques et trigonométriques, il y a une certaine gamme de fonctions qui peuvent être utilisées pour exécuter une variété d'opérations sur les chaînes de caractères. Pour les détails, référez-vous à la description des commandes.

11 Fichiers

Un fichier est un ensemble de données structuré qui peut être conservé dans une unité extérieure (comme l'unité de disques par exemple). L'ensemble de données peut être lu dans la zone de travail pour le traitement et écrit à nouveau sur le fichier selon la nécessité en utilisant un simple identificateur: soit le nom de fichier.

Un fichier peut contenir les données de texte en forme de codes représentatifs de caractères dans les lignes d'un programme BASIC, ainsi que les données binaires. Les fichiers peuvent être stockés sur le disque RAM, la bande cassette ou le disque souple et échangés entre différents dispositifs. Ils peuvent aussi être envoyés, à travers les ports série d'E/S, à ou par autres ordinateurs.

Tous les fichiers dans le PC-1600 sont des fichiers séquentiels, c'est-à-dire que les données sont écrites et lues du fichier de manière séquentielle, article par article. Un accès aléatoire aux données au milieu du fichier n'est pas possible dans le PC-1600.

Descripteurs de fichier

Les fichiers sont identifiés par un "descripteur". Le descripteur du fichier spécifie le nom de dispositif (d:), le nom de fichier et l'extension (.ext) par le format suivant:

d: nom de fichier.ext

Nom de dispositif

Le nom de dispositif spécifie sur quel dispositif le fichier doit être stocké en tant que nouveau fichier ou sur quel dispositif Je fichier existe. Le nom de dispositif doit être suivi d'un deux-points, à savoir :. A titre d'exemple, les noms de dispositifs suivants sont utilisés:

Nom	Dispositif
S1 :	Module de mémoire dans le logement 1
S2:	Module de mémoire dans le logement 2
COM1 :	Port série RS-232C
COM2:	Port série optique
COM:	Port série en cours d'utilisation

CAS:	Lecteur à cassette connecté à l'unité traceur/imprimante CE-1600P
X:, Y:	Unité de disques souples de 2,5" optionnelle

Nom de fichier

Tous les fichiers doivent avoir un nom pour la sauvegarde dans un dispositif (seule exception faite des fichiers stockés sur la bande cassette en mode compatible avec PC-1500 par l'instruction CSAVE). Le nom de fichier peut être spécifié sans aucune contrainte par huit caractères au maximum, y compris les caractères suivants:

A à Z, 0 à 9, #, $, %, &, ' , ( ), −, ^, { }, @

Un seul nom peut être attribut à un fichier.

Extension

L'extension est une façon supplémentaire d'identifier le type de fichier (par exemple, le fichier de programmes BASIC, le fichier de programmes en langage machine, le fichier de texte).

L'extension est composée de trois caractères au maximum ajoutés après le nom de fichier avec un point intercalé. L'utilisateur peut spécifier n'importe quelle extension valable à un fichier lors de sa sauvegarde. A titre d'exemple, tous les fichiers de programmes relatifs à la comptabilité peuvent avoir une extension, .FIN. Par ailleurs, l'extension .BIN pourrait être utilisée pour tous les fichiers binaires en langage machine. Il est facile d'identifier les contenus du fichier en regardant le répertoire d'un disque.

Lorsqu'un fichier de programme BASIC est sauvegardé dans un dispositif par la commande SAVE, le PC-1600 affecte automatiquement une extension à trois lettres, à savoir .BAS, au nom du fichier pour l'identifier comme programme BASIC. Lorsque le fichier est chargé à nouveau dans la mémoire par la commande LOAD, il n'est pas nécessaire de spécifier l'extension .BAS; l'ordinateur peut alors deviner que l'extension est .BAS, même si elle n'est pas spécifiée.

Lorsque la commande FILES ou LFILES est utilisée pour lister le répertoire des fichiers sur un dispositif, les fichiers de programmes BASIC apparaissent sur la liste imprimée avec l'extension .BAS, à moins qu'une certaine autre extension n'ait été spécifiée lors de leur sauvegarde. L'extension .BAS doit être toujours spécifiée en cas d'utilisation de la commande COPY.

Stockage et extraction du fichier

Les fichiers peuvent être stockés sur une bande cassette, un disque souple et un disque RAM.

Fichiers sur bande cassette

Les fichiers sont séquentiellement stockés sur bande cassette, c'est-à-dire que les fichiers sont retenus sur la bande dans l'ordre de stockage. Ceci signifie que, pour obtenir le cinquième fichier stocké sur la bande, il est nécessaire d'enrouler la bande d'une longueur correspondant aux quatre premiers fichiers à compter de sa tête.

Chaque fichier sur la bande possède une étiquette portant le nom de fichier. Si vous spécifiez la commande LOAD pour charger un fichier de la bande, l'ordinateur contrôle le défilement de la bande de lecteur de façon à retrouver l'étiquette portant le nom de fichier ayant été spécifié en introduisant la commande LOAD et le charge. Comparée avec le disque souple ou le disque RAM, cette méthode nécessite plus de temps pour rechercher le fichier y étant stocké.

Les commandes suivantes sont disponibles pour créer, rappeler et mettre à jour les fichiers chargés sur la bande cassette:

BLOAD, BSAVE, CHAIN, CLOAD, CLOAD?, CLOAD M, CLOSE, CSAVE, CSAVE «M, INPUT#, LOAD, MAXFILES, MERGE, OPEN, PRINT#, RMT ON/OFF, SAVE

Les commandes CLOAD, CLOAD?, CLOAD M, CSAVE et ÇSAVE M sont pour une compatibilité descendante avec l'ordinateur de poche PC-1500 et n'ont aucune fonction spéciale pour le PC-1600.

Le nombre de fichiers pouvant être stockés sur une bande cassette est déterminé par la longueur de la bande et la taille des fichiers eux-mêmes.

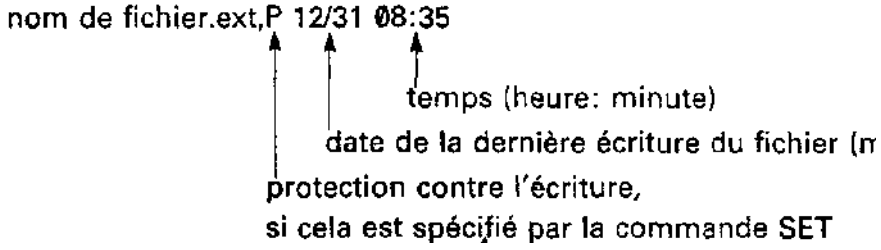
Fichiers sur disque

Les fichiers sont stockés sur le disque souple ou RAM en secteurs miniatures qui peuvent être positionnés n'importe où sur le disque (ou en RAM). L'ordinateur es1 capable d'accéder à un point quelconque sur le disque (ou dans la RAM), d'une façon à la fois directe et immédiate. Ceci signifie que tous fichiers stockés sur le disque souple ou le disque RAM peuvent être instantanément chargés dans la mémoire de l'ordinateur.

Répertoire

Pour savoir où en est le fichier sur le disque, l'ordinateur tient un répertoire de tous les fichiers et leurs tailles, avec autres informations nécessaires pour l'accès.

Le répertoire est affiché sur l'écran soit pour le disque souple, soit pour le module RAM, à l'aide de la commande FILES ou sorti sur l'imprimante à l'aide de la commande LFILES. (Cette fonction est indisponible pour la bande cassette, car les fichiers y étant stockés ne sont pas répertoriés.) Le répertoire contient les informations sur le ou les fichiers spécifiés selon le format suivant:


nom de fichier.ext,P 12/31 08:35
temps (heure: minute)
date de la dernière écriture du fichier (m
protection contre l'écriture,
si cela est spécifié par la commande SET

Les commandes suivantes sont disponibles pour créer, rappeler ou mettre à jour les fichiers stockés sur le disque souple ou le disque RAM:

BLOAD, BSAVE, CLOSE, COPY, FILES, INIT, INPUT#, KILL, LFILES, LOAD, MAX-FILES, NAME, OPEN, PRINT#, SAVE, SET

Nombre de fichiers par dispositif

Le nombre de fichiers qui peuvent être stockés sur un disque souple ou RAM dépend de la taille du bloc répertoire; de façon concise, le nombre de fichiers pouvant être répertoriés sur un disque souple est de 48. Le nombre maximum de fichiers pouvant être retenus dans le module RAM optionnel CE-1600M est de 48.

REMARQUE: Le module de programmes CE-159 ne peut être utilisé seulement que pour le stockage d'un seul programme, lequel n'est cependant pas sous la forme du fichier portant son nom. Après que le module a été inséré dans l'un des logements à l'arrière de l'ordinateur, le logement considéré est rendu actif par la commande TITLE. Le programme dans ce module est alors exécuté par introduction de RUN comme s'il était chargé dans la mémoire intérieure de

Noms de fichier ambigus - "Wildcards"

Un groupement de fichiers peut être spécifié par un nom de fichier ambigu qui est satisfait par tous les noms de fichiers du groupement. Ceci est effectué en utilisant les caractères "wildcard" pour certains caractères de la chaîne constitutive du nom de fichier ambigu. Le caractère "wildcard" est soit le point d'interrogation ? qui

peut remplacer un nombre quelconque de caractères simples composant le nom de fichier, soit l'astérisque * qui peut remplacer n'importe quel nom de fichier et/ou n'importe quelle extension dans leur ensemble.

Exemple:

Nom de fichier ambigu	Noms de fichiers correspondant au nom ambigu
?ET	SET, MET, RET, (5ET, 4ET
?PT?	RPT1, RPT2, SPTM, (PT), -PTS
?????	Tous les noms de fichiers à cinq caractères
???1.B??	REP1.BAS, DAT1.BIN, LET1.BKK
*	Tous les noms de fichiers à 8 caractères
* .*	Tous les noms de fichiers avec une extension quelconque

Lorsqu'un nom de fichier ambigu est spécifié avec des caractères "wildcard" à l'aide de la commande FILES ou LFILES, tous les fichiers dont le nom a trait au nom ambigu sont listés sur l'écran ou l'imprimante. Le format * . * permet de lister tous les fichiers répertoriés dans le dispositif sur l'écran ou l'imprimante. La description de la commande FILES dans le répertoire des commandes porte également sur l'utilisation des caractères "wildcard" * et ? dans la spécification des noms de fichiers. Le caractère "wildcard" ? peut aussi être utilisé pour programmer l'heure de signalisation sonore avec l'instruction ALARM$.

Protection du fichier

Chacun des fichiers stockés sur le disque souple ou le disque RAM peut être protégé avec la commande SET en BASIC. Une fois protégé, le fichier ne peut plus accepter l'écriture, ni être détruit. Pour les détails concernant les types de protection, référez-vous à la description de la commande SET.

De plus, les fichiers pourront être protégés de manière matérielle. Ceci se matérialise, pour le disque souple, par une languette de protection contre l'écriture, qui située au bord du disque, protège tous ses contenus contre l'effacement ou l'édition. Pour le disque RAM, les contenus peuvent être protégés en mettant au travail l'interrupteur de protection en écriture placé sur la face frontale du module. Dans ce cas, l'ordinateur doit être hors tension. Les contenus du disque sont alors protégés contre l'effacement ou l'édition.

De même, les contenus de la mémoire peuvent être protégés contre tout accès non autorisé en spécifiant un mot de passe à l'aide de la commande PASS. Une fois protégés, ils ne peuvent plus être effacés, édités ou listés, à moins que l'utilisateur n'entre le mot de passe correct.

Création d'un fichier

Pour la création d'un fichier, il faut suivre les quatre étapes suivantes dans l'ordre correct:

1 Exécutez l'instruction MAXFILES pour définir le nombre de fichiers devant être ouverts.

2Exécutez l'instruction OPEN pour définir le nom du fichier et le numéro correspondant, puis ouvrir le fichier en mode OUTPUT.

3Ecrivez les données sur le fichier avec l'instruction PRINT#.

4 Exécutez l'instruction CLOSE pour fermer le fichier. Cette instruction doit être spécifiée avant de rouvrir le fichier pour entrer les données en mode INPUT.

Exemple:

Le court programme donne ci-dessous permet de créer, sur le disque souple, un fichier dont les données nom et adresse peuvent servir de base pour un petit carnet d'adresses informatisé.

```
  5 MAXFILES = 1                              Affecte 1 au nombre de fichiers
                                                  devant être ouverts.
 10 OPEN "X:ADDRESS" FOR OUTPUT AS #1         Ouvre le fichier pour la sortie.
 20 INPUT "ENTER NAME?";N$                    Stocke les données concernant le
 30 IF NS = "END" THEN 100                    nom de ville et le numéro de
 40 INPUT "ENTER CITY?";C$                        téléphone, introduites par le
 50 INPUT "ENTER TEL NUMBER?";T$                  clavier.
 60 PRINT #1, N$;",";C$;",";T$                Ecrit les données sur le fichier 1.
 70 PRINT                                     Curseur à la ligne suivante
 80 GOTO 20                                   Entre les données consécutives.
100 CLOSE #1                                  Ferme le fichier.
110 END
```

Entrez le programme dans le PC-1600 et tapez RUN. Le message de sollicitation suivant apparaît alors sur l'écran:

ENTER NAME?

Tapez le nom d'une personne dont vous désirez stocker l'adresse et le numéro de téléphone, puis appuyez sur la touche ENTER.Maintenant, vous aurez sur l'écran le message de sollicitation suivant:

ENTER CITY?

Tapez le nom de la ville et appuyez sur ENTER. Le message de sollicitation suivant apparaîtra.

ENTER TEL NUMBER?

Tapez le numéro de téléphone et appuyez sur la touche ENTER. Le premier message de sollicitation apparaît à nouveau pour vous inciter à entrer les trois données sur la deuxième personne. A partir des données suivantes, créez un fichier exemple d'adresses:

M.JONES	NEW YORK	212-758-0354
T.SMITH	LONDON	01-634-4431
M.BERRY	NEW YORK	212-432-0012
N.ITO	TOKYO	03-927-1345
S.SHARP	NEW YORK	212-124-5364
P.PETERS	LONDON	01-433-0056

Lorsque vous aurez terminé l'entrée de toutes les données, tapez END en réponse au caractère de sollicitation ENTER NAME?. Ainsi, le programme est terminé et le fichier de données, fermé.

Si vous regardez le répertoire du disque souple avec la commande FILES en utilisant le nom de fichier ambigu * . *, vous pouvez confirmer que le fichier "ADDRESS" est ajouté au répertoire par ce programme. Ce fichier contient toutes les données introduites dans le programme dans la même séquence d'entrée.

IMPORTANT: Si vous déroulez le programme encore une fois, le fichier ADDRESS est rouvert pour la sortie à la ligne 70. *Le procédé de réouverture du fichier pour la sortie efface toutes les données précédemment stockées et laisse un fichier vide, appelé adresse, prêt à l'écriture.*

La seule méthode d'addition d'une nouvelle information aux données existant dans un fichier est de rouvrir ce dernier en mode APPEND. Un programme séparé doit être créé à cet effet avec le fichier ouvert, par l'instruction suivante:

OPEN "X:ADDRESS" FOR APPEND AS #1

Pour un fichier séquentiel, les données peuvent seulement être ajoutées en fin du fichier. Ceci signifie que les articles ne peuvent pas être insérés en position alphabétique avec APPEND. Pour résoudre ce problème, il suffit cependant d'écrire un autre programme qui permet de faire lire à l'ordinateur toutes les données du fichier, de mettre les additions en position alphabétique correcte et ensuite de récrire l'ensemble de données sur le fichier dans le nouvel ordre. APPEND ne peut être utilisé que pour les fichiers résidant sur le disque souple ou dans le module RAM.

Accès au fichier

Pour l'accès au fichier, il faut suivre les quatre étapes suivantes dans l'ordre correct:

1 Exécutez l'instruction MAXFILES pour désigner le nombre de fichiers devant être ouverts.

2 Exécutez linstruction OPEN pour ouvrir le fichier "X:ADDRESS" en mode INPUT.

3 Faites tire à l'ordinateur les données du fichier pour les mettre en mémoire de variables à l'aide de l'instruction INPUT#.

4 Exécutez l'instruction CLOSE pour fermer le fichier. Ceci est à exécuter avant que le fichier ne soit rouvert en mode OUTPUT ou APPEND.

Chaque fois que le fichier est ouvert, les, données sont lues dès le début. A chaque exécution de l'instruction INPUT#, les données suivantes dans le fichier sont stockées dans le programme. Lorsque le fichier arrive à sa fin et qu'il n'y a aucune instruction dans le programme pour arrêter la lecture du fichier, le code d'erreur 165 est affiché.

Exemple:

Le programme indiqué ci-dessous permet de lire les données stockées sur le fichier ADDRESS créé avec le programme précédemment décrit et d'afficher sur l'écran toutes les personnes habitant à NEW YORK:

```
  5 MAXFILES = 1

 10 OPEN "X:ADDRESS" FOR INPUT AS #1
 20 PRINT "NEW YORK"
 30 PRINT:PRINT "NAME", "TEL NUMBER"
 40 IFEOF(1)THEN 100
 50 INPUT #1,N$, C$, T$
 60 IF C$ = "NEW YORK" THEN 80
 70 GOTO 40
 80 PRINT N$, T$
 90 GOTO 40 100
CLOSE #1 110
END
```

Désigne le nombre de fichiers devant être ouverts. Ouvre le fichier pour l'entrée. Imprime le nom de ville et les en-têtes. Contrôle la fin du fichier. Lit les articles du fichier. Compare la ville. Revient pour lire l'article suivant Imprime le nom et le numéro. Revient pour lire l'article suivant, Ferme le fichier.

Entrez le programme dans le PC-1600 et tapez RUN. Alors, sur l'écran apparaissent les informations suivantes:

NEW YORK

NAME	TELNUMBER
M.JONES	212-758-0354
M.BERRY	212-432-0012
S.SHARP	212-124-5364

Pour la comparaison dans la ligne 60 pour la chaîne "NEW YORK", il faut que l'orthographe et l'espacement de la chaîne soient exactement les mêmes avec les lettres majuscules.

Ce programme peut être facilement modifié pour permettre à l'utilisateur de choisir la ville à rechercher, en ajoutant ou remplaçant les lignes suivantes:

```
15 INPUT "WHICH CITY";S$
20 PRINT S$
60 IF C$ = S$ THEN 80
```

Mise à jour du fichier

Comme il a été précédemment décrit, les données ne peuvent être ajoutées ou annexées qu'en fin du fichier lorsque celui-ci est rouvert en mode APPEND. Pour insérer les articles dans un fichier existant, un programme plus sophistiqué est requis pour lire toutes les données et les mettre à jour dans la mémoire de PC-1600 et pour les récrire sur le fichier dans un nouvel ordre.

L'information supplémentaire sur le nom peut être ajoutée à la queue du fichier ADDRESS existant en remplaçant la ligne 10 du premier programme comme suit:

```
10 OPEN "X:ADDRESS" FOR APPEND AS #1
```

Les commandes DSKF, EOF, LOC, LOF et MAXFILES sont disponibles pour un programme de traitement de fichiers plus évolué afin de vérifier la taille de chaque fichier ou cours de sa création et de s'assurer de l'absence d'erreur au niveau du dispositif et des fichiers.

12 Accès aux ports série

Deux ports série de communication intégrés au PC-1600 permettent de connecter celui-ci avec autres ordinateurs et dispositifs servant à soutenir l'accès à l'interface RS-232C ou ayant un port à fibres optiques compatible. Les deux ports, RS-232C et optique, sont traités par BASIC comme dispositifs d'entrée/sortie séquentielle et identifiés par les noms de dispositif, à savoir COM1: et COM2:. COM: spécifie le port en vigueur.

Spécification du port

Les deux ports sont ouverts pour la communication de données par l'instruction SETDEV. Une fois spécifié COM:1 ou COM2: avec SETDEV, le port considéré est utilisé pour l'entrée et la sortie de commandes BASIC tout au long de la communication, à moins qu'une autre commande SETDEV ne soit exécutée. Pour la gestion des deux ports série, les commandes suivantes sont disponibles:

COMn ON/OFF/STOP, PCONSOLE, INIT, INSTAT, ON COMn GOSUB, ON PHONE GOSUB, OUTSTAT, PHONE ON/OFF/STOP, RCVSTAT, SETCOM, SETDEV, SNDBRK, SNDSTAT, PZONE

De plus, certaines commandes d'impression peuvent être utilisées pour l'entrée par l'un des deux ports série. Suite à la mise sous tension, la sortie par les commandes LLIST, LFILES et LPRINT est normalement dirigée à l'imprimante. La commande SETDEV possède trois options qui conduisent la sortie de ces commandes au port sélectionné. SETDEV est aussi utilisé pour renvoyer la sortie à l'imprimante, fermant effectivement les deux ports série. L'accès en vigueur pour la commande SETDEV peut être affiché en utilisant la fonction DEV$. Les commandes SAVE et LOAD sont disponibles pour envoyer ou recevoir l'ensemble de de fichiers par l'intermédiaire des ports série.

Options de protocole

Avant la réception ou la transmission des données par un port série, le protocole de communication régissant les états de signaux de commande doit être choisi selon le dispositif avec lequel le PC-1600 entre en communication. Le protocole de liaison est spécifié avec la commande SNDSTAT pour la transmission et avec la commande RCVSTAT pour la réception. Pour la transmission, les états de tous les signaux de commande peuvent être individuellement spécifiés avec la commande OUTSTAT.

La spécification du protocole nécessite une connaissance approfondie dans le domaine des signaux de commande utilisés pour la transmission de données; toutefois, le PC-1600 est réglé pour la transmission simple de données si les paramètres OUTSTAT ne sont pas spécifiés. Le protocole de réception en vigueur peut être affiché en utilisant la commande INSTAT.

Paramètres de communication

Pour la communication avec un autre ordinateur ou dispositif, le débit et le mode de transmission de données doivent être compatibles entre les deux unités. Les paramètres programmables sont les suivants:

1Débit de transmission (débit en bauds) variable de 50 à 38400 bauds
2Longueur de mot variable de 5 à 8 bits
3Parité, telle que paire, impaire, nulle
4Nombre de bits de stop, 1 ou 2
5protocole X-ON/OFF, valable ou non
6protocole de décalage, valable ou non

Ces paramètres sont spécifiés par la commande SETCOM. Les paramètres en vigueur peuvent être affichés par la fonction COM$.

En cas d'utilisation des commandes SAVE, LOAD, BSAVE, BLOAD ou de la commande COPY pour le transfert de fichiers entre le disque RAM et l'un quelconque des ports, le débit en bauds maximum est fixé à 9600 bauds pour l'accès RS-232C (COM1:) et à 38400 bauds pour le port optique (COM2:). D'autre part, en cas d'utilisation des commandes INPUT, INPUT#, PRINT#, LLIST et LPRINT, le débit en bauds est limité à 4800 bauds pour les deux ports.

Tampon de réception

L'information reçue par l'un des deux ports série est d'abord stockée dans le tampon de réception. La taille de ce tampon peut être définie par l'utilisateur avec la commande INIT de manière appropriée au type de l'information devant être reçue.

Sortie à un port série

Les commandes et fonctions pouvant être utilisées pour la sortie à l'un des ports série sont les suivantes:

CHR$, LFILES, LLIST, LPRINT/LPRINT USING, OPEN, PCONSOLE, PRINT#/PRINT# USING, PCONSOLE, PZONE, SAVE

Transmission des programmes et données

Les données produites par un programme ou les listes de programmes peuvent être transmises directement à un port série en procédant comme suit:

1Ouvrez le port avec l'instruction SETDEV en spécifiant l'option PO pour sortir les commandes LPRINT, LLIST et LFILES à cet accès.

2Utilisez la commande PZONE pour définir le format de sortie des commandes LPRINT.

3 Utilisez la commande PCONSOLE pour définir la longueur appropriée de ligne et la fin du code ligne.

Ainsi, le fichier de programme peut être transmis à un autre ordinateur. D'abord, chargez le fichier dans la mémoire à l'aide de la commande LOAD. Ensuite, conduisez la sortie au port série RS-232C en utilisant la commande SETDEV. Enfin, tapez LLIST pour sortir les lignes de programme, une par une, au port série.

Pour le fichier et les données séquentiellement écrits, le port peut aussi être ouvert avec l'instruction PRINT# en procédant comme suit:

1Définissez le nombre de fichiers devant être ouverts, en utilisant la commande MAXFILES.

2Ouvrez le port pour la sortie et spécifier son numéro avec la commande OPEN.

3Utilisez la commande PRINT# ou RPINT# USING dans un programme pour écrire les données élémentaires à destination du port de la même façon que pour un fichier ordinaire.

4Fermez le port avec l'instruction CLOSE comme pour un fichier ordinaire.

Transmission des fichiers de programme

Le fichier existant sur un disque souple, un disque RAM ou une bande peut être directement transmis à l'un des ports série à l'aide de la commande SAVE dans un but de sauvegarde comme pour le disque souple ou RAM.

Transmission des codes de contrôle

La fonction CHRS peut être utilisée pour envoyer les codes de caractères spéciaux à l'un des ports de la même façon que pour leur envoi à une imprimante. Par exemple, le code de contrôle ASCII 4 est utilisé, dans nombreux ordinateurs, pour dire le fin du texte (EOT). Ce code peut être transmis au port série en spécifiant l'instruction du format suivant après avoir ouvert le port pour la sortie à l'aide de SETDEV.

```
10 LPRINT CHR$(4)
```

Entrée par un port série

Les commandes et fonctions pouvant être utilisées pour l'entrée par l'un des ports série sont les suivantes:

INIT, !NPUT#, LOAD, OPEN, PCONSOLE/RXD$, SNDBRK

Réception des données

Les données reçues par l'un des ports série peuvent être stockées dans un programme en procédant comme suit:

1Définissez le nombre de fichiers à ouvrir à l'aide de la commande MAXFILES.

2Définissez la taille du tampon de réception selon la nécessité par la commande INIT.

3Ouvrez le port pour l'entrée du fichier par la commande OPEN.

4Stockez les données du port en utilisant la commande INPUT# de la même façon que pour un fichier ordinaire.

5Fermez le port avec l'instruction CLOSE.

Réception des fichiers

Le fichier peut être reçu par l'intermédiaire d'un port série et stocké dans la mémoire en spécifiant la commande LOAD comme pour charger un fichier du disque souple ou RAM.

13 Mise au point

Il est pratiquement impossible d'écrire un programme impeccable en une seule fois. Des fautes de frappe ou erreurs de syntaxe (erreurs de grammaire) peuvent se glisser au cours de la programmation, lorsque vous spécifiez vos instructions en BASIC. Même un contrôle méticuleux suivant l'entrée ne permettra pas de trouver toutes les erreurs, particulièrement celles au niveau de la conception logique d'un programme. La mise au point signifie le procédé d'élimination. Systématique de différents types d'erreurs permettant d'obtenir le programme conforme aux conditions de la conception.

Erreurs de syntaxe

Lorsque vous exécutez votre programme pour la première fois, des erreurs, à commencer par les erreurs de syntaxe, sont détectées, ligne par ligne, au fur et à mesure du déroulement du programme.- L'exécution est arrêtée à la première erreur qui se glisse dans le programme et le code d'erreur et le numéro de ligne correspondants sont affichés sur l'écran.

Dans ce cas, vous devez vous référer à la liste des codes d'erreurs et, pour leur signification, à l'annexe F. Si l'erreur détectée est une erreur de syntaxe (code d'erreur 1), tapez la commande LIST <numéro de ligne> pour lister la ligne comportant l'erreur. Ou, alternativement, appuyez sur la touche CL pour effacer le code d'erreur, puis sur la touche MODE pour changer le mode PRO et puis sur la
touche ↓.La ligne sera affichée avec le curseur indiquant l'erreur. Utilisez ensuite les touches d'édition pour corriger l'erreur et tapez encore une fois RUN pour relancer le programme. L'ordinateur arrête l'exécution du programme lorsqu'une erreur suivante est détectée. Répétez cette procédure chaque fois que l'exécution est interrompue par le code d'erreur 1.

Les autres types d'erreurs doivent être traités selon le cas. Certaines erreurs peuvent se glisser dans votre conception logique, alors qu'autres sont faciles à corriger; par exemple, spécifier une commande FOR..TO sans définir l'instruction NEXT correspondante. A travers une pratique intensive, vous pourrez établir la meilleure approche systématique pour la mise au point de vos programmes au-delà de l'étape de syntaxe.

Les erreurs de syntaxe et les fautes de frappe peuvent être corrigées grâce aux fonctions d'édition décrites à la section "Mode d'édition" dans le chapitre 9. Vous trouverez qu'au bout de la pratique, les lignes peuvent être corrigées avec le minimum de frappe de ces touches.

Mode d'analyse

Les commandes TRON (mode d'analyse au travail) et TROFF (mode d'analyse au

repos) peuvent être utilisées pour suivre l'exécution d'un programme, ligne par ligne.

Lorsque le mode d'analyse est activé par la commande TRON, l'ordinateur exécute une ligne de programme et arrête l'exécution 0,5 seconde en affichant le numéro de ligne correspondante à droite de l'écran, avant de passer à la ligne suivante. Ce mode est utile pour l'analyse du déroulement de votre programme. Assurez-vous que le programme saute aux endroits corrects au bon moment et ainsi de suite.

Le mode d'analyse peut aussi être spécifié pour exécuter une ligne simple; dans ce cas, pour passer à la ligne suivante, pressez sur la touche de flèche orientée vers le bas, soit ↓. Cette façon d'analyse vous donne un temps plus ample pour examiner la séquence d'exécution du programme. Le mode d'analyse persiste jusqu'à ce que la commande TROFF soit exécutée. Pour de plus amples détails, référez-vous à la description sur le répertoire des commandes.

Les commandes STOP peuvent être insérées, de manière stratégique, dans votre programme en phase de mise au point pjour arrêter l'exécution par l'ordinateur à un certain point et imprimer les résultats intérimaires de contrôle. L'exécution du programme peut ensuite être reprise au même point en utilisant la commande CONT.

Programmes de traitement d'erreurs

Il est possible d'empêcher les messages d'erreur d'interrompre l'exécution du programme, s'ils sont dûs à une situation prévisible. Si, par exemple, vous répondez à l'instruction INPUT en tapant une fausse donnée dans votre programme. Utilisez la commande ON ERROR GOTO ensemble avec les fonctions ERL et ERN pour continuer l'exécution du programme.

La commande ON ERROR GOTO est spécifiée au début du programme et signale où se trouve la première ligne du programme de traitement d'erreur. Après l'exécution de cette instruction, toutes les erreurs dans le programme causent le branchement avec la première ligne du programme de traitement d'erreur, lequel est conçu soit pour corriger l'erreur, soit pour vous permettre de retaper une information correcte, avec pour résultat de relancer le programme sans qu'il ne soit interrompu par le message d'erreur.

Dans le cadre du programme, les fonctions ERL et ERN sont utilisées pour déterminer de quel type d'erreur s'agit-il et sur quelle ligne se trouve-t-elle dans le programme. Le programme de traitement d'erreur permet de reprendre l'exécution du programme au point considéré avec l'instruction RESUME ou de terminer l'exécution avec un message convenable pour l'utilisateur.

Pour un exemple simplifié du programme de traitement d'erreur, se référer à la description de la commande ON ERROR GOTO.

14 Dictionnaire des commandes BASIC

Le Dictionnaire des commandes comprend toutes les commandes et fonctions BASIC valables pour le PC-1600. Certaines commandes sont incluses pour assurer une parfaite compatibilité avec le PC-1500 en MODE 1 et ne sont pas donc si utiles en mode PC-1600.

Les symboles énumérés ci-dessous sont utilisés tout au long de la description des commandes pour identifier les modes dans lesquels elles peuvent être spécifiées, ainsi que les dispositifs périphériques nécessaires.

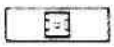
Peut être directement introduit en mode PRO

Peut être directement introduit en mode RUN

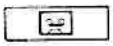
Utilisé pour programmer les touches de fonction en mode RESERVE.

Peut être introduit comme ligne de programme.

Commande de disque souple. L'unité à disques de poche, CE-1600F, doit être en connexion.

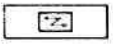


Commande de bande cassette. Un enregistreur à cassette doit être connecté à travers l'interface cassette CE-1600P.



Commande d'imprimante. Le PC-1600 doit être installé dans l'interface imprimante/cassette CE-1600P (ou dans le CE-150).

Commande de communication. Utilisé avec l'un des deux ports série. Un dispositif sériel adéquat doit être connecté au port série spécifié.



Commande de disque RAM. Le module de programme CE-1600M ou CE-161 doit être installé et initialisé dans la fente 1 ou 2.

Les symboles suivants sont utilisés dans le format de commande tout au long de la description des commandes:

[ ] Le paramètre mis entre crochets carrés est optionnel. Les crochets eux-mêmes ne font pas partie de la commande à entrer.

< > Utilise pour indiquer la valeur en vigueur du paramètre à entrer. Les crochets eux-mêmes en font pas partie de la commande à entrer.

( ) Utilisé pour délimiter la valeur du paramètre dans certaines commandes et doit être entré en tant qu'une partie de commande.

ABS

FORMAT: 1. ABS(X)

Abréviation: AB *Voir également:*

BUT: Renvoie la valeur absolue du nombre X.

REMARQUES: Le nombre X est une expression numérique quelconque

EXEMPLE:

```
5:WAIT 70
10:PRINT "          NOMBRE          ABSOLUE
20:FOR I=-2 TO 2
30:PRINT I ,ABS(I)
40:NEXT I
50:END

>RUN
```

>

NOMBRE	ABSOLUE
-2	2
-1	1
0	0
1	1
2	2

ACS

FORMAT: 1. ACS(X)

Abréviation: AC.
Voir également: ASN, ATN, COS

BUT: Renvoie l'arc cosinus de X.

REMARQUES: Cette fonction renvoie le cosinus inverse de l'expression en degrés, radians ou gradients selon laquelle le mode de l'ordinateur est spécifié par la commande de DEGREE, RADIAN ou GRAD. La valeur de X est limitée à $-1 \leqq X \leqq 1$.

EXEMPLE: En mode DEG, ACS(X),est renvoyée dans une gamme de 0 à 180°. En mode RAD, ACS(X) est renvoyée dans une gamme de 0 à *n* radians. En mode GRAD, ACS(X) est renvoyée dans une gamme de 0 à 200.

```
10:DEGREE
20:PRINT "L'ARC COS DE 0.5 EST";ACS (0.5)
30:PRINT "L'ARC COS DE 0 EST";ACS (0)
40:END

>RUN
L'ARC COS DE 0.5 EST 60
L'ARC COS DE 0 EST 90
>
```

ADIN ON/OFF/STOP

FORMAT: 1. ADIN ON
2.ADIN OFF
3.ADIN STOP

Abréviation: AD.
Voir également: AIN, ON ADIN GOSUB

BUT: Valide ou invalide les interruptions reçues par l'intermédiaire du plot d'entrée analogique.

REMARQUES: Les interruptions sont reçues par le plot d'entrée analogique sous forme de tension d'un certain niveau. Pour les détails des niveaux de tension applicable, voir la description de l'instruction AIN.

ADIN ON valide les interruptions reçues par l'intermédiaire du plot analogique. L'instruction ON ADIN GOSUB peut alors être utilisée pour exécuter par branchement un programme correspondant à l'interruption. ADIN OFF invalide les interruptions provenant du plot d'entrée analogique. ADIN STOP invalide les interruptions provenant du plot d'entrée analogique mais enregistre la demande d'interruption reçue en dernier. Si ADIN ON est exécutée subséquemment, cette demande d'interruption sera immédiatement réalisée. La valeur implicite (valeur par défaut) est ADIN STOP.

EXEMPLE: Voir ON ADIN GOSUB.

AIN

FORMAT: 1. AIN

Abréviation: AI.
Voir également: ADIN ON/OFF/STOP, ON ADIN GOSUB

BUT: Renvoie une valeur correspondant au niveau d'entrée de tension analogique.

REMARQUES AIN est une variable de système qui maintient le résultat de conversion du niveau de tension d'entrée analogique en niveau numérique dans une gamme de 0 à 255. En ce qui concerne les détails des niveaux de tension admissibles pour le signal d'entrée analogique, voyez le chapitre 6.

EXEMPLE:

```
>PRINT AIN
43
>
```

Imprime la tension analogique exprimée en valeur numérique

ALARM$

FORMAT: 1. ALARM$="MM/DD/HH/mm[;<chaîne de messages>]
2.ALARM$=""
3.ALARM$

Abréviation: AL.
Voir également: TIMES, DATE$, POWER

BUT: Spécifie l'heure d'alarme et une chaîne optionnelle de messages

REMARQUES: ALARM$="MM/DD/HH/mm" désigne l'heure d'alarme, où,

MM: mois	(01 à 12)	' Séparés par une barre oblique (/)
DD: jour HH:	{01 à 31)	
heure	(00 à 23)	
mm: minute	(00 à 59)	

A l'heure spécifiée, l'alarme interne de l'ordinateur retentir avec des bips répétés pendant 1 seconde. Des caractères facultatifs "?" peuvent être utilisés dans la spécification du mois et jour pour désigner l'alarme mensuelle ou journalière. Par exemple, "??/??/13/30" permet à l'alarme de retentir journellement à 13:30.

La <chaîne de messages> optionelle peut être d'une longueur de 26 caractères au maximum. Le message sera affiché sur l'écran lorsque le l'heure d'alarme est atteint. ALARM$= " " supprime et désigne à nouveau l'heure d'alarme et la chaîne de messages optionnelle. ALARM$ utilisée comme une variable renvoie l'heure d'alarme actuellement désigné.

NOTE:
La chaîne de messages spécifiée recouvrira le contenu de. la chaîne de caractères des touches de fonction pour mode II. Pour les détails, voir le chapitre 9, Mode de réserve.

EXEMPLE:

```
>ALARM$="25/12/08/00;JOYEUX NOEL!"
```

Cela imprimera le message "JOYEUX NOËL!" à 8 heures le jour Noël

AREAD

FORMAT: 1. <étiquette>[:]AREAD<nom de variable>

Abréviation: A.
Voir également: RUN

BUT: Permet à l'article affiché sur l'écran d'être introduit dans une variable de programme.

REMARQUES: La commande AREAD ne peut être utilisée qu'immédiatement après l'étiquette sur la première ligne d'un programme et est effective seulement si l'exécution du programme est lancée par l'enfoncement de la touche DEF suivi par l'étiquette. Dans tous les autres cas, AREAD sera ignorée.

Lorsque AREAD est effective, elle prend les chiffres ou caractères affichés sur l'écran au moment où elle est exécutée et, les attribue à la variable spécifiée dans le <nom de variable>. Le nom de variable doit correspondre au type d'information: une variable numérique pour chiffres et une variable alphanumérique pour caractères. Si l'affichage contient des chiffres, 10 chiffres et 2 exposants au maximum peuvent être introduits dans la variable. Si l'affichage contient des caractères, le nombre de caractères pouvant être introduits est égal à la longueur de la variable alphanumérique comme déclaré dans l'instruction DIM. La valeur implicite est de 16 caractères. Si seulement le caractère de sollicitation de BASIC (>) est affiché lorsque AREAD est exécuté, le contenu du <nom de variable> sera remis à zéro.

EXEMPLE:

```
10 : "A" : AREAD N
```

[10] Lorsque le programme est exécuté, le chiffre, (s'il ya), actuellement affiché sur l'écran sera introduit dans la variable de programme "N".

ARUN

FORMAT: 1. ARUN

; ARU.

Voir également: RUN

BUT: Permet à l'ordinateur de commencer automatiquement l'exécution d'un programme par suite de la mise sous tension.

REMARQUES Lorsque ARUN est incluse en tant que première instruction de programme (c'est-à-dire avec le numéro de ligne le plus jeune dans le programme) lors de la mise sous tension, le programme sera exécuté comme lorsqu'une commande RUN a été introduite à partir du clavier. L'ordinateur doit être mis hors tension en mode RUN. ARUN est ignorée si elle n'est pas la première instruction.

NOTE:
Si des unités périphériques quelconques (imprimante, enregistreur à cassette, unité de disque souple et modules de mémoire) sont remplacées pendant que l'alimentation est coupée, l'ordinateur peut renvoyer un code d'erreur pour indiquer que les unités périphériques ne sont pas prêtes à fonctionner et n'exécutent pas le programme. Dans ce cas, lancer l'exécution avec la commande RUN. L'exécution d'ARUN par suite de la mise sous tension ne supprime pas de variables ou paramètres comme la commande RUN le fait. Utiliser l'instruction CLEAR dans le programme pour le faire.

EXEMPLE:

```
10:ARUN :CLS
20:PRINT "BIENVENUE DANS LE MONDE DU"
30:PRINT "PC-1600"
40:PRINT "IL EST MAINTENANT";TIME$
50:PRINT "VOUS DISPOSEZ DE";MEM;"OCTETS LIBRES"
60:END
```

Le programme est automatiquement déroulé à la mise sous tension.

ASC

FORMAT: 1. ASC {<variable alphanumérique>) 2.
ASC ("<chaîne>")

Abréviation:
Voir également: CHR$

BUT: Revoie l'élément de code ASCII pour le premier caractère dans la chaîne spécifiée.

REMARQUES La chaîne peut être spécifiée comme contenu d'une variable alphanumérique sous forme de X$ ou comme chaîne actuelle bornée entre guillement, "XXXX". Seule la valeur du premier caractère dans la chaîne est renvoyée indépendamment de la longueur. Pour les tables de code de caractères, voir l'annexe C.

EXEMPLE:

```
10:INPUT "INTORODWIRE UN CARACTERE";A$
20:N=ASC (A$)
30:PRINT "LE CODE ASCII EST";N
40:GOTO 10
50:END
```

[10] L'utilisateur appuie sur une touche pour introduire un caractère quelconque.
[20] ASC trouve le numéro de code pour ce caractère.
[30] Imprime la réponse.
[40] Répète jusqu'à ce que l'utilisateur interrompt le programme en appuyant sur la touche de rupture.

ASN

FORMAT: 1. ASN(X)

Abréviation: AS.

Voir également: ACS, ATN, SIN

BUT: Renvoie l'arc sinus de X.

REMARQUES: Cette fonction renvoie le sinus inverse de l'expression en degrés, radians ou gradients selon lequelle le mode de l'ordinateur est spécifié par la commande de DEGREE, RADIAN ou GRAD. La valeur de X est limitée à -1=X^1.

EXEMPLE: En mode DEG, ASN(X) est renvoyée dans une gamme de -90 à 90°. En mode RAD, ASN(X) est renvoyée dans une gamme de -*ττ/2* à *τî/2* radians. En mode GRAD, ASN(X) est renvoyée dans une gamme de -100 à 100.

```
5:WAIT 60
10:CLS
20:DEGREE
30:PRINT "ARC SIN","ANGLE"
40:FOR H=0 TO 10
50:X=H/10
60:DX=ASN X
70:PRINT X,DX
80:NEXT H
```

ATN

FORMAT: 1. ATN(X)

Abréviation: AT.
Voir également: ACS, ASN, TAN

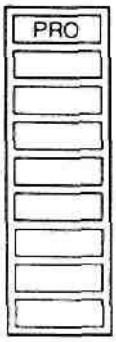
BUT: Renvoie l'arc tangente de X.

REMARQUES Cette fonction renvoie la tangente inverse de l'expression en degrés, radians ou gradients selon laquelle le mode de l'ordinateur est spécifié par la commande DEGREE, RADIAN ou GRAD.

EXEMPLE: En mode DEG, ATN(X) est renvoyée dans une gamme de -90 à 90°. En mode RAD, ATN(X) est renvoyée dans une gamme de -xr/2 à 7r/2 radians. En mode GRAD, ATN(X) est renvoyée dans une gamme de -100 à 100.

```
10:CLS
20:RADIAN
25:PRINT"TANGENTE","ANGLE"
30:FOR T=0 TO 20
40:RT=ATN (T)
50:PRINT T,RT
60:NEXT T
```

AUTO



FORMAT:

1.AUTO
2.AUTO <numéro de ligne>
3.AUTO <numéro de ligne>, <incrément>
4.AUTO, <incrément>

Abréviation: AU.
Voir également: RE N U M

BUT: Lance la génération automatique du numéro de ligne de programme pour faciliter l'introduction au clavier du programme en mode PRO.

REMARQUES Une fois AUTO introduite, chaque fois que la touche ENTER est enfoncée après l'introduction au clavier d'une ligne de programme, la ligne d'entrée suivante apparaît avec le curseur mis à la position prête à l'introduction après le numéro de la nouvelle ligne. Si une instruction existe déjà en mémoire avec ce numéro de ligne, la ligne existante est affichée.

L'introduction de AUTO sans aucun paramètre engendre les numéros de ligne commençant à 10 par incréments de 10. L'introduction de AUTO <numéro de ligne> engendre les numéros de ligne depuis le numéro spécifié à des incréments de 10. L'introduction de AUTO <numéro de ligne>, <incrément> engendre les numéros de ligne à partir du numéro spécifié à des incréments spécifiés.

Pour annuler la commande AUTO, appuyer sur la touche **BREAK** ou la touche **CL**, ou enfoncer la touche à **ENTER** deux reprises.

EXEMPLE:

```
>AUTO
```

Engendre les numéros de ligne 10, 20, 30, 40 ....

```
>AUTO 100
```

Engendre les numéros de ligne 108, 110, 120, 130 ...

```
>AUTO 400,20
```

Engendre les numéros de ligne 400, 420, 440, 460 ....

BEEP

FORMAT: 1. BEEP <nombre>
2. BEEP <nombre>[,<ton>[,<durée>]]

Abréviation: B. *Voir également:* ON/OFF

BUT: Engendre les bips à une tonalité et une durée spécifiées par l'intermédiaire du haut-parleur intégré à l'ordinateur.

REMARQUES: Le paramètre <nombre> spécifie le nombre d'émissions du signal sonore dans une gamme de 0 à 65535.

L'option <ton> spécifie l'élévation de la tonalité (fréquence) du bip dans une gamme de 255 à 0. 255 correspond à une fréquence de 230 Hz environ, 0 à 7 kHz environ. La valeur par défaut pour <ton> donne une fréquence de 4 kHz environ.

L'option <durée> spécifie la durée du bip. La consigne de la durée du bip varie selon le paramètre <ton>. Même si une valeur <durée> est spécifiée, la durée est d'autant plus longue que la fréquence est plus basse. La valeur par défaut est 160.

EXEMPLE:

```
10:FOR I=1 TO 3
20:FOR J=50 TO 250 STEP 50
30:BEEP I,J
40:NEXT J
50:NEXT I
60:END
```

[10] Cette boucle externe est utilisée pour changer le nombre de bips de 1 à 3.

[20] Le compteur de boucle interne est utilisé pour changer la tonalité.

[30] L'instruction BEEP est exécutée 15 fois. Noter que la durée du bip devient plus longue suivant l'abaissement du ton même si la valeur par défaut de <durée> est utilisée chaque fois.

BEEP ON/OFF

FORMAT: 1. BEEP ON
2. BEEP OFF

Abréviation: B.
Voir *également:* BEEP

BUT: Invalide ou valide à nouveau la commande BEEP.

REMARQUES: BEEP OFF invalide la commande BEEP. Elle invalide également les tonalités de contrôle produites lorsque l'ordinateur est en train de lire un fichier en cassette.

BEEP ON valide à nouveau l'instruction BEEP après la désignation de BEEP OFF.

EXEMPLE:

```
10:BEEP 4
20:BEEP OFF
30:BEEP 4
40:BEEP ON
50:END
```

[101 Emet le signal sonore 4 fois.
[20] Toutes les instructions subséquentes BEEP seront invalidées jusqu'à ce que BEEP ON soit exécutée.
[30] Lorsque le programme est déroulé, seuls 4 bips de ligne 10 sont entendus.
[40] En principe, il convient de rétablir les valeurs par défaut pour le logiciel. Ici, BEEP est validée à nouveau à la fin du programme.

BLOAD

FORMAT: 1. BLOAD "<d:nom de fichier>"[,#<bloc de mémoire>#<adresse>]

Abréviation: BL.
Voir également: BSAVE, CLOAD, NEW, SET

BUT: Charge un programme de langage machine dans la mémoire à partir du disque souple, du module de disque RAM ou d'une cassette.

REMARQUES: Le paramètre <d:nom de fichier> spécifie l'unité et le nom de fichier. Les unités suivantes peuvent être spécifiées:

S1:, S2:	Modules de disque RAM
X:, Y:	Unité de disque souple
COM1 :, COM2:	RS-232C et accès séquentiel E/S optique
CAS:	Bande cassette

Le paramètre <bloc de mémoire> spécifie un bloc de mémoire de 0 à 7 en tant que destination pour le programme de langage machine.

Le paramètre <adresse> spécifie une adresse dans le bloc de mémoire spécifié en système hexadécimal. Il s'agit de l'adresse à partir de laquelle le stockage du programme commencera. Si les deux paramètres ne sont pas spécifiés, le programme sera remis à l'adresse dans le bloc de mémoire depuis laquelle il a été conservé dans l'unité.

Si le programme est sauvegardé à l'origine en spécifiant une adresse de lancement automatique (voir l'instruction BSAVE), il sera chargé et exécuté automatiquement à partir de cette adresse. Pour les détails concernant les blocs de mémoire et les adresses, voir l'annexe D, blocs de mémoire et adresses. *EXEMPLE:*

```
>BLOAD"X:RXOUT"
```

Charge le programme de langage machine RXOUT sur l'unité de disque souple X: dans la même adresse de mémoire depuis lequel il a été conservé.

BREAK ON/OFF

FORMAT: 1. BREAK ON
2. BREAK OFF

Abréviation: BR.
Voir également: CONT

BUT: Utilisée pour invalider ou valider à nouveau la touche BREAK.

REMARQUES BREAK OFF invalide la touche BREAK. Cela signifie qu'un programme ne peut être interrompu à partir du clavier pendant l'exécution. BREAK ON valide à nouveau la touche BREAK. Durant l'exécution du programme, si la touche BREAK est enfoncée, l'exécution estjnterrompue, et le message "BREAK IN <numéro de ligne>" est affiché sur l'écran. L'exécution peut être reprise avec la commande CONT.

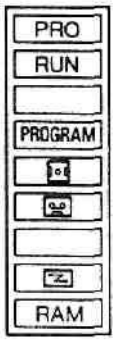
En principe, il convient de désigner BREAK OFF lors du lancement des programmes longs avec un petit affichage sur l'écran ou impression; ordinairement les programmes de calcul. Cependant, BREAK ON doit être spécifiée à la fin du programme pour valider à nouveau la touche BREAK pour usage normal. Si un programme est interrompu dans une boucle infinie lorsque la touche BREAK est mise hors fonction, l'exécution ne peut être arrêtée que par l'enfoncement de la touche RESET. Le mieux est donc de mettre complètement au point les programmes avant incorporation de l'instruction BREAK OFF.

EXEMPLE:

```
>BREAK OFF
>RUM
>BREAK ON
```

RUN lance l'exécution d'un programme long. Afin d'éviter toute interruption du programme par enfoncement de la touche de rupture, BREAK OFF est introduite au clavier avant le passage du programme. A la fin du programme, BREAK ON est introduite au clavier pour valider à nouveau la touche de rupture. BREAK OFF et BREAK OFF pourraient être alternativement incluses dans le programme même, repectivement près du début et de la fin, permettant ainsi d'éviter l'introduction au clavier de BREAK ON à la fin du programme.

BSAVE



FORMAT: 1. BSAVE "<d:nom de fichier>",#<bloc de mémoire>,
Odresse de lancement>,<adresse de fin>
[,<adresse de lancement automatique>]

Abréviation: BS.
Voir également: BLOAD, CSAVE M

BUT: Conserve un programme de langage machine sur un disque souple ou module de disque RAM

REMARQUES: Le paramètre <d:nom de fichier> spécifie l'unité et le nom de fichier. Les unités suivantes peuvent être spécifiées:

S1:, S2:	Disques RAM
X:, Y:	Unité de disque souple
COM1 :, COM2:	Ports E/S série optique et RS-232C
CAS:	Bande cassette

Si l'unité spécifiée est protégée en écriture, la commande engendrera un code d'erreur.

Le paramètre <bloc de mémoire> spécifie un bloc de mémoire de 0 à 7 où le programm de langage machine est stocké.

Le paramètre odresse de lancement> spécifie l'adresse inférieure dans le bloc de mémoire spécifié où le programme est stocké.

Le paramètre odresse de fin> spécifie l'adresse supérieure dans le bloc de mémoire spécifié où le programme est stocké.

Le paramètre odresse de lancement automatique> spécifie l'adresse depuis laquelle l'auto-exécution doit être mise en marche après que le programme a été chargé à la même adresse de mémoire avec la commande BLOAD. Lorsqu'aucun paramètre Odresse de lancement automatique> n'est spécifié, la valeur implicite &FFFF arrête la fonction de lancement automatique.

Pour les détails concernant les blocs de mémoire et les adresses, voir l'annexe D.

A la différence de la commande SAVE, aucun nom d'extension de fichier n'est automatiquement ajouté au nom de fichier.

EXEMPLE:

```
>BSAVE"S1:TRIAGE ",#1.& 3000,&3AFF
```

Conserve le programme de langage machine dans le bloc de mémoire 1 depuis l'adresse &8000 jusqu'à l'adresse &8AFF sur le disque RAM dans le logement 1 avec nom de fichier TRIAGE.

CALL

FORMAT: 1. CALL [#<bloc de mémoire>,]<adress>[,<nom de variable>]

Abréviation: CA.

Voir également: NEW, POKE, XPOKE

BUT: Utilisée pour appeler et exécuter un programme de langage machine à partir d'un programme BASIC.

REMARQUES: L'instruction CALL appelle un programme de langage machine ou un sous-programme stocké en mémoire. Lorsque le programme est appelé, une variable simple peut être passée depuis le programme BASIC en cours au programme de langage machine, et lorsque L'exécution du programme de langage machine est terminée, la même variable simple sera renvoyée au programme d'appel. Le registre B et la paire de registres DE sont utilisés dans ce but. Le programme de langage machine doit être introduit de mémoire en utilisant l'instruction POKE ou XPOKE avant qu'il ne puisse être appelé.

Le paramètre <bloc de mémoire> spécifie un bloc de mémoire de 0 à 7 où le programme de langage machine est stocké. Si ce paramètre n'est pas spécifié, le bloc de mémoire implicite est bloc 0.

Le paramètre <adresse> spécifie l'adresse inférieure dans le bloc de mémoire spécifié où le programme de langage machine est stocké dans une gamme de 0 à 65535 (&0 à &FFFF).

Le paramètre <nom de variable> spécifie la variable dont la valeur doit être passée au programme de langage machine à l'entrée, et renvoyé au programme d'appel BASIC à la sortie si le drapeau de report est élevé. Si le paramètre <nom de variable> spécifie une variable numérique, la valeur doit être un nombre entier dans une gamme de -32768 à 32767. La valeur est passée à la paire de registres DE à l'entrée et les contenus de la paire de registres DE sont renvoyés au programme d'appel comme valeur BCD avec le même nom de variable à la sortie. Si le paramètre <nom de variable> spécifie une variable alphanumérique, la paire de registres DE maintient l'adresse de lancement du lieu de chaîne, et le registre B maintient la longueur de la chaîne.

EXEMPLE:

```
400:CALL #3,&8000,X
410:PRINT "LA VALEUR DE X RENVOYEE"
420:PRINT "DU LANGAGE MACHINE EST";X
```

[400] Dirige les passages au langage machine maintenu dans le bloc de mémoire 3 avec lancement d'exécution à l'adresse de mémoire &8000.
[410-420] La commande CALL est passée à travers la variable numérique "X" et ia valeur renvoyée est imprimée ici.

CHAIN

FORMAT: 1. CHAIN
2. CHAIN [<nom de fichier>"][,<numéro de ligne>]

Abréviation: CHA.
Voir également: CSAVE, MERGE

BUT: Permet à un programme BASIC de charger et d'exécuter un programme sur la bande cassette.

REMARQUES: Le programme à chaîner doit être sur la bande.

La commande CHAIN charge le premier programme sur bande dans la mémoire et L'exécute à partir du premier numéro de ligne. CHAIN <numéro de ligne> charge le programme sur bande dans la mémoire et l'exécute à partir du numéro de ligne spécifié. CHAIN <nom de fichier> recherche le programme appelé <nom de fichier> sur la bande, l'introduit en mémoire et l'exécute à partir du premier numéro de ligne. Si l'option <numéro de ligne> est également spécifiée, l'exécution est mise en marche à partir de ce numéro de ligne.

L'utilisation de l'instruction CHAIN permet aux programmes plus grands que la capacité de la mémoire d'être divisés en sous-programmes qui tiendront dans la mémoire, pour être chargés et exécutés d'une façon séquentielle. Si un mot de passe est désigné avec la commande PASS, CHAIN engendrera une erreur.

EXEMPLE:

```
>RUN

10:REM VOICI LE PROG1
20:
30:
:
:
400:CHAIN "PROG2"

10:REM VOICI LE PROG2
20:
30:
:
:
200:END
>
```

RUN lance l'exécution de PROG1 qui se trouve actuellement dans la mémoire. La ligne finale [400] est enchaînée au programme de continuation PROG2 conservé sur la bande cassette. PROG2 est chargé à partir de la bande dans la mémoire et l'exécution se continue.

CHR$

FORMAT: 1. CHR$ (<expression nombre entier>) 2.
CHR$ (<nombre entier>)

Abréviation: CH. *Voir également:* ASC

BUT: Renvoie le caractère dont le code ASCII est égal à la valeur de l'<expression nombre entier> ou la valeur du <nombre entier>.

REMARQUES La fonction CHR$ peut être utilisée pour envoyer des caractères de commande spéciaux au terminal ou à un port série, ou pour imprimer les caractères graphiques qui ne sont pas sur le clavier à partir d'un programme .BASIC. Pour les tables de de code caractère, voir l'annexe C.

EXEMPLE:

```
10:FOR X=33 TO 126
20:PRINT CHR$(X);
30:NEXT X
40:END
```

[20] Imprime les caractères de table de code ASCII sur l'écran.

CLEAR

FORMAT: 1. CLEAR

Abréviation: CL.

Voir également DIM, ERASE, TITLE

BUT: Efface toutes les variables de la mémoire, y compris les variables fixes.

REMARQUES CLEAR libère la place utilisée par toutes les variables comprenant les variables fixes A-Z, A$-Z$, ou @(1)-@(26) qui ne nécessitent pas d'instruction DIM, car elles sont affectées en permanence, et remet les valeurs numériques à zéro et les variables alphanumériques à nulle chaîne (code ASCII 0).

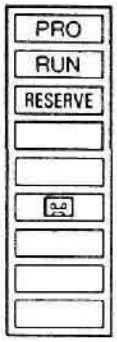
La commande CLEAR peut être également utilisée dans un programme. Elle est utilisée pour récupérer la place occupée dans la mémoire afin de stocker les variables lorsqu'elles ne sont plus nécessaires; par exemple, lorsque les variables utilisées dans la première partie d'un programme ne sont pas nécessaires dans la seconde partie et que l'espace mémoire disponible est limité. CLEAR peut être également utilisée au début d'un programme lorsque certains programme sont résidants en mémoire et que vous voulez libérer l'espace mémoire utilisé par exécution de programmes précédents.

EXEMPLE:

```
5:WAIT 30
10:DIM C(5)
20:FOR N=1 TO 5
30:READ A:LET C(N)=A:PRINT C(N)
40:NEXT N
50:DATA 10,20,30,40,50
60:CLEAR
70:PRINT A
```

[51 Spécifie le temps d'attente pour afficher l'impression.
[10] Dimensionne un tableau C (N).
[20-40] Introduit les nombres de la ligne de donnée dans les éléments de C (N) et imprime.
[60] Libère la place affectée à C(1) à C(5) et remet A à 1.
[70] Imprime la valeur effacée pour A.

CLOAD



FORMAT: 1. CLOAD ["<nom de fichier>"][,R]

Abréviation: CLO.
Voir également: CLOAD?, CSAVE, MERGE

BUT: Utilisée pour charger dans la mémoire un programme ou un programme d'utilisation de touche de fonction conservé par l'instruction CSAVE en mode PC-1500.

REMARQUES CLOAD efface toutes les lignes de programme en cours d'exécution dans la mémoire et charge le premier programme à partir de la bande commençant par la position de bande actuelle. CLOAD "<nom de fichier>" efface toutes les lignes de programme en cours d'exécution dans la mémoire, explore la bande pour rechercher le programme avec le nom spécifié et le charge dans la mémoire.

En modes PRO et RUN, le programme est chargé dans une mémoire, tandis qu'en mode RESERVE, le programme d'utilisateur de touche de fonction est chargé dans une mémoire de réserve. Il est nécessaire de veiller à ce que les programmes ne soient pas chargés dans une mémoire inapropriée par erreur, en vue d'éviter une destruction des contenus en cours.

Si l'option R est spécifiée, le fichier sera chargé dans la mémoire et exécuté automatiquement comme si RUN est introduit. Si le fichier ne contient pas de programme BASIC, une erreur d'exécution aura lieu.

Si le mot de passe est désigné avec la commande PASS, CLOAD engendrera une erreur.

CLOAD chargera seulement les programmes qui ont été conservés sur la bande à l'aide de la commande CSAVE.

EXEMPLE:

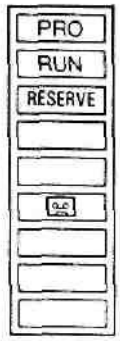
```
>CLOAD
```

Charge dans la mémoire le premier programme trouvé sur la bande à partir de la position de tête de bande en cours.

```
>CL0AD"PROG0 1"
```

Recherche sur la bande le programme "PROG01" à partir de la position de tête de bande en cours et le charge dans la mémoire.

CLOAD?



FORMAT: 1. CLOAD? ["<nom de fichier>"]

Abréviation: CLO.?
Voir également: CLOAD, CSAVE, MERGE

BUT: Utilisée pour vérifier qu'un programme a été enregistré sur la bande cassette sans aucune erreur en mode PC-1500.

REMARQUES: Le programme vérifié peut être celui utilisé à la fois en mode RUN et en mode PRO, ou en mode RESERVE le programme d'utilisateur de touche de fonction.

Lors de l'utilisation de la commande CLOAD? seule, la bande doit être manuellement rembobinée au commencement du programme avant l'introduction de la commande. Le programme sera alors lu à nouveau de la bande et comparé bloc par bloc avec la version dans la mémoire. S'il y a une erreur quelconque, un code d'erreur sera engendré. S'il n'y a aucune erreur, le caractère de sollicitation (>) réapparaîtra.

CLOAD? "<nom de fichier>" explore la bande pour rechercher le programme spécifié et le vérifier avec les contenus de mémoire.

EXEMPLE:

```
>CL0AD
>CL0AD?
```

Charge et vérifie le premier programme trouvé sur la bande à partir de la position de tête de bande cours.

```
>CLOAD"PROG0 1" >CL0
AD?"PROG0 1"
```

Recherche sur la bande le programme "PROG01" à partir de la position de tête de bande en cours, le charge et le vérifie.

CLOAD M

FORMAT: 1. CLOAD M ["<nom de fichier>"][;#<bloc de mémoire>,<adrésse>]

Abréviation: CLO. M
Voir également: BLOAD, CALL, CSAVE M, NEW

BUT: Utilisée pour charger dans la mémoire un programme de langage machine sur bande cassette en mode PC-1500.

REMARQUES Des programmes de langage machine sont chargés dans une zone de mémoire autre que celle pour les programmes normaux BASIC. Et, le format utilisé pour leur conservation sur la bande est différent. Un programme de langage machine ne peut être chargé avec la commande CLOAD"normale.

Le paramètre <bloc de mémoire> spécifie un bloc de mémoire de 0 à 7 comme la destination pour le programme de langage machine.

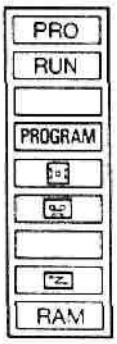
Le paramètre <adresse> spécifie une adresse dans le bloc de mémoire spécifié en système hexadécimal. Il s'agit de l'adresse à partir de laquelle le stockage du programme sera commencé. Si ces deux paramètres ne sont pas spécifiés, le programme sera chargé à nouveau à l'adresse dans le bloc de mémoire depuis lequelle il a été sauvegardé dans l'unité. Si le programme est sauvegardé à l'origine avec une adresse de lancement automatique spécifiée (voir la commande CSAVE M), il sera automatiquement chargé depuis cette adresse et exécuté.

EXEMPLE:

```
>CLOADM"MAC 1
```

Charge le programme de code machine MAC1 sur bande dans le même bloc de mémoire et l'adresse depuis laquelle il a été stocké avec CSAVE M, Les informations sur le bloc de mémoire et l'adresse sont maintenues avec programme de langage machine sur bande.

CLOSE



FORMAT: 1. CLOSE
2. CLOSE #<N° de fichier>[,#<N° de fichieo ....]

Abréviation: CLOS.
Voir également: END, OPEN

BUT: Termine l'accès à un fichier sur l'unité en cours de sollicitation.

REMARQUES: La commande CLOSE spécifiée, seule, ferme tous les fichiers ouverts. Le paramètre <n° de fichier> est le numéro sous lequel le fichier a été ouvert avec l'instruction OPEN.

Une fois ouvert, un fichier doit être fermé avant qu'il ne puisse être ouvert à nouveau dans un but différent (c'est-à-dire l'entrée, sortie ou addition). Si une tentative est faite pour ouvrir un fichier qui est déjà ouvert, un code d'erreur sera engendré.

Tous les fichiers sont automatiquement fermés lors de l'exécution des commandes END, NEW, RUN et LOAD, lorsque le programme est mis en forme ou que l'ordinateur est mis hors tension.

EXEMPLE:

```
5:.MAXFILES =2
IO:ÛPEN "X:PAIEMENT*FOR INPUT AS #I
20:0PEN "CAS:.MISE A JOUR"FOR OUTPUT AS *2


400:CLOSE *1,#2
```

[400] Ferme les deux fichiers après exécution.

CLS

FORMAT: 1. CLS

Abréviation: Voir également:

BUT: Efface l'écran.

REMARQUES: CLS efface toutes les lignes de l'écran et met le curseur à la position d'origine (0,0) en MODE 0 en haut à gauche de l'écran.

EXEMPLE:

```
>CLS
```

Efface l'écran et remet le curseur à la position d'origine.

COLOR

FORMAT: 1. COLOR <numéro>

Abréviation: COL.
Voir également:

BUT: Désigne la couleur de stylo pour l'imprimante.

REMARQUES: Quatre couleurs peuvent être spécifiées par l'instruction COLOR

0-noir 1-
bleu 2-
vert 3-
rouge

Lors de la mise sous tension, la couleur par défaut est le noir (0)

EXEMPLE:

```
>C0LOR 3
```

Choisit le stylo rouge.

COM$

FORMAT: 1 COM$ "COMn:"

Abréviation: COM.
Voir également: SETCOM

BUT: Renvoie une chaîne contenant les valeurs désignées pour les paramètres de communication pour le port spécifié dans la commande SETCOM.

REMARQUES: La chaîne contient les paramètres dans le même ordre comme spécifié dans la commande SETCOM: <BR>, <WL>, <PR>, <ST>, <XO>, <SI>.

COM$ "COM1:" renvoie les valeurs pour le port série RS-232C. COM$ "COM2:" renvoie les valeurs pour le port série E/S optique. COM$ "COM:" renvoie les valeurs pour le port en cours d'ouverture.

EXEMPLE:

```
10:SETCOM "COM1:",300,3,M,1.X.S
20:PRINT COMS'CÛMI:*
```

(10] Désigne les paramètres pour l'interface série RS-232C.
[20] Imprime la chaîne contenant les paramètres désignés dans la ligne 10.

COMn ON/OFF/STOP

FORMAT: 1. COMn ON
2.COMn OFF
3.COMn STOP

u Abréviation:

Voir également: ON COMn GOSUB, SETCOM

BUT: Valide ou invalide les interruptions reçues par l'intermédiaire des ports de communication.

REMARQUES: Le paramètre n spécifie le port RS-232C (n=1) ou le port E/S optique (n = 2).

COMn ON valide les interruptions à recevoir par l'intermédiaire du port spécifié. L'instruction ON COMn GOSUB peut être alors utilisée pour exécuter par branchement un programme correspondant à l'interruption. COMn OFF invalide les interruptions provenance du port spécifié. COMn STOP invalide les interruptions provenant du port spécifié, mais retient la plus récente des demandes d'interruption reçues. Si COMn ON est subséquem-ment exécutée, cette demande d'interruption sera immédiatement réalisée. La valeur implicite est COMn STOP.

EXEMPLE:

```
ÎOICOMI  ON
20: 30:
40:COM1  OFF
```

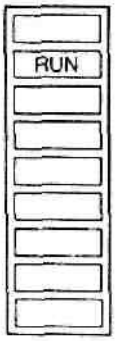
[10] Valide les interruptions au port de RS-232C. [40| Invalide les interruptions au port de RS-232C,

FORMAT: 1. CONT

Abréviation; C.
Voir également: RESUME, RUN, STOP, WAIT

BUT:

CONT



Reprend l'exécution d'un programme qui a été interruompu par une instruction STOP, une instruction PRINT ou une touche BREAK.

REMARQUES: Cette commande est souvent utilisée pendant la mise au point d'un programme avec STOP. Lorsque l'exécution est interrompu par STOP ou en appuyant sur la touche BREAK, l'introduction de CONT au clavier reprendra l'exécution du programme au point où il a été terminé, à condition qu'aucun changement de programme ne soit effectué durant l'interruption. (La commande GOTO <numéro de ligne> peut être introduite au clavier pour résumer l'exécution à un numéro de ligne différent.) Noter que CONT ne reprendra pas l'exécution après END ou lorsque l'exécution est interrompue par un cas d'erreur. Pour cela, voir RESUME.

Après que ARRET termine le programme, CONT est utilisée pour continuer le déroulement du

EXEMPLE:

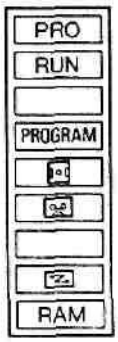
```
10:PRINT "LE PROGRAMME S'ARRETE ICI"
20:STOP
30:PRINT "APRES ARRET(STOP)"
40:PRINT "AU REVOIR"
50:END

>RUN
LE PROGRAMME S'ARRETE ICI

 BREAK IN 20
>
>CONT
APRES ARRET
AU REVOIR
>
```

programme depuis une ligne suivant la ligne où la rupture a eu lieu.

COPY



FORMAT: 1. COPY "<d1:nom de fichier1.ext>" TO "<[d2:j nom de fichier2.ext>"

Abréviation: COP. *Voir également:* SET

BUT: Copie un fichier depuis une unité à une autre unité.

REMARQUES: La commande COPY peut être utilisée pour copier des fichiers entre deux unités. Le paramètre <d1:nom de fichier1.ext> spécifie l'unité et le nom de fichier pour copier depuis la (source). Le paramètre <d2:nom de fichier2.ext> spécifie l'unité et le nom de fichier pour copier à la (destination). L'extension du fichier doit être toujours donnée lors de la reproduction d'un fichier.

Si l'unité destination n'est pas spécifiée, l'unité source est utilisée. La commande COPY ne peut être utilisée pour copier depuis un port série à un disque ou une bande.

Le fichier ne sera pas copié si le nom de fichier2 existe déjà sur l'unité de destination.

La commande COPY peut être également utilisée pour changer le nom d'unité de disque souple. Le nom d'unité est initialement spécifié à X: Cependant, PC-1600 peut adresser le nom d'unité Y: Cela permet des copies de sauvegarde de vos disques avec l'unité simple. Pour copier d'un disque à un autre, voir le paragraphe concernant l'unité de disque souple au chapitre 6.

EXEMPLE:

```
>COPY"S1:RICH.BAS* TO *s2:RICH.BAS
```

Copie le fichier RICH.BAS depuis le module de disque RAM dans le logement d'extension 1 au même nom de fichier dans le module de disque RAM dans le logement d'extension 2.

```
>COPY*X:HAC.BAS* TO "HACOPY.BAS*
```

Copie le fichier HAC.BAS sur un disque souple sur un nouveau fichier HACOPY.BAS sur le même disque souple.

FORMAT: 1. COS(X)

Abréviation:
Voir également: ACS, SIN, TAN

COS

BUT: Renvoie le cosinus de l'angle X.

REMARQUES: Cette fonction renvoie le cosinus de l'angle X; X est exprimé en degrés, radians ou grandients. Selon cette expression, le mode de l'ordinateur est spécifié par la commande DEGREE, RADIAN ou GRAD.

EXEMPLE:

```
10:DEGREE
20:PRINT "LE COS DE 60 EST "COS(60)
30:PRINT "LE COS DE 90 EST "COS(90)
40:END

>RUN
LE COS DE 60 EST 0.5
LE COS DE 90 EST 0
>
```

CSAVE

FORMAT: 1. CSAVE
2. CSAVE ["<nom de fichier>"][,A][;<numéro de ligne>,<numéro de ligne>]

Abréviation: CS.
Voir également: CLOAD, CLOAD?, LLIST, MERGE

BUT: Sauvegarde un programme ou une partie d'un programme mémorisés, sur la bande cassette en mode PC-1500.

REMARQUES : Le programme conservé deviendra un programme normal en mode RUN ou PRO, ou un programme d'utilisateur de touche de fonction en mode RESERVE.

CSAVE en elle-même conserve le programme entier commençant par la position de bande en cours, et sans nom de programme spécifié.

CSAVE "<nom de fichier>" conserve le programme sous le nom de fichier spécifié. Lorsque l'option [,A] est spécifiée, le fichier est conservé en format ASCII; autrement il est conservé en forme binaire. L'option <numéro de ligne>, <numéro de ligne> permet une spécification d'une séquence quelconque de numéros de ligne de la même manière que pour, la commande LLIST.

Si un mot de passe est spécifié par la commande PASS, le programme ne peut être conservé par CSAVE avant que le mot de passe ne soit supprimé.

Lors du chargement d'un programme qui a été conservé sur la bande par la commande CSAVE, la commande CLOAD doit être utilisée.

EXEMPLE:

```
>CSAVE"PROG01";200,330
```

Sauvegarde les lignes 200 à 380, du programme en cours d'exécution, en mémoire sur la bande sous le nom de fichier "PROG01".

CSAVE M

FORMAT: 1. CSAVE M "<nom de fichier>";#<bloc de mémoîre>,
odresse de lancement>,<adresse de fin>
Lodresse de lancement automatique>]

Abréviation: CS. M
Voir également: CLOAD M, CALL, BSAVE

BUT: Sauvegarde un programme de langage machine mémorisé, sur la bande cassette en mode PC-1500.

REMARQUES: Le paramètre <bloc de mémoire> spécifie un bloc de mémoire de 0 à 7 où le programme de langage machine est stocké. Le paramètre odresse de lancement> spécifie l'adresse inférieure dans le bloc de mémoire spécifié où le programme est stocké. Le paramètre odresse de fin> spécifie l'adresse supérieure dans le bloc de mémoire spécifié où le programme est stocké. Le paramètre odresse de lancement automatique> est optionnel et spécifie l'adresse à partir de laquelle l'exécution doit être lancée après rechargement. La valeur implicite est &FFFF, qui bloque la fonction de lancement automatique.

EXEMPLE:

```
>CSAVEM"MAC 1";*2,&3000,&3AFF
```

Sauvegarde le programme de langage machine "MAC1" sur la bande qui est actuellement affecté dans le bloc de mémoire 2 depuis («adresse &800O à &8AFF (32768 à 35583). Les informations sur le bloc de mémoire et l'adresse sont stockées avec le programme lui-même sur la bande. Si CLOAD M est utilisée ultérieurement pour recharger le programme dans la mémoire sans spécifier explicitement ces valeurs, les valeurs sur la bande seront automatiquement lues et utilisées.

CSIZE

FORMAT: 1. CSIZE <taille>

Abréviation: CSI.
Voir également: PCONSOLE

BUT: Spécifie la taille des caractères sortis sur papier.

REMARQUES: Le paramètre <taille> spécifie la taille de caractère dans une gamme de 1 à 9. La taille de caractère spécifiée pour la valeur du paramètre <taille> est donnée dans le tableau suivant:

<taille> consigne	1	2	3	4	5	6	7	8	9
Cara./ligne	160	80	53	40	32	26	22	20	17
Hauteur (mm)	1,2	2,4	3,6	4,8	6,0	7,2	8,4	9,6	10,8
Largeur (mm)	0,8	1,6	2,4	3,2	4,0	4,8	5,6	6,4	7,2

NOTE:
Les tailles de caractère correspondant aux valeurs de paramètre <taille> données dans le tableau ci-dessus sont avec la longueur de ligne spécifiée à l'infini avec l'instruction PCONSOLE, <longueur>=0. *EXEMPLE:*

```
>CS I ZE  2
```

Spécifie la taille des caractères à imprimer à la consigne 2.

CURSOR

FORMAT: 1. CURSOR <colonne> [,<ligne>]

Abréviation: CL).
Voir également: PRIN T

BUT: Met le curseur à une position quelconque sur l'écran.

REMARQUES La commande CURSOR positionne le curseur sur une ligne quelconque dans une gamme de 0 à 25 prête à afficher les données. La ligne est spécifiée dans une gamme de 0 (sommet) à 3 (fond). La valeur implicite correspond à la ligne actuelle sur laquelle est mis le curseur.

EXEMPLE:

```
10:WAIT 50
20:FOR N=1 TO 6
30:READ A$
40:CURSOR 12,1
50:PRINT A$
60:NEXT N
70:END
80:DATA "B","O","N","J","O","U","R","!"
```

Ce programme affichera le message Bonjour! au milieu de l'écran avec chaque caractère subséquent recouvrant le caractère précédent.

[10] WAIT crée un retard court d'impression de caractères de afin que la séquence soit facile à lire.
[20] Tourne sur une boucle des 6 caractères.
(30] Lit le caractère suivant.
[40] Déplace le curseur à la position de caractère 12 le long de la seconde ligne à partir du sommet de l'écran.
[50] Imprime le caractère et recouvre tout caractère existant.

DATA

FORMAT: 1. DATA liste des constantes

Abréviation: DA.
Voir également: READ, RESTORE

BUT: Liste les données élémentaires à mémoriser par l'instruction READ.

REMARQUES: Une instruction DATA peut contenir des constantes numériques ou alphanumériques quelconques séparées par des virgules. Les constantes de chaîne doivent être comprises entre guillements.

Une fois que les données élémentaires sur une ligne de données ont été mémorisées, elles ne peuvent être stockées jusqu'à ce qu'une instruction RESTORE soit exécutée.

Les instructions de données ne sont pas exécutables et peuvent donc être rangées à n'importe quelle partie d'un programme, souvent comme les dernières lignes où elles peuvent être facilement trouvées pour la modification ultérieure. Un programme peut contenir un nombre quelconque de lignes de données, et celles-ci seront mémorisées dans l'ordre de numéro de ligne par les instructions READ. Une seule instruction READ n'a pas besoin de correspondre à une seule instruction DATA.

EXEMPLE:

```
10:FOR J=1 TO 4
20:READ A$,B
30:PRINT A$,B
40:NEXT J
50:END
60:DATA "DANIEL:DANY",23"DAVID",38,"ALINE",-24,"    CLAUDE",34

>RUN
DANIEL:DANY  23
DAVID         38
ALINE        -24
    CLAUDE    34
>
```

DATE$

FORMAT: 1. DATES-"MM/DD" 2.
DATES

Abréviation: DATE.
Voir également: TIMES, ALARM$

BUT: La commande DATES est une variable de système qui contient la date de l'horloge en temps réel, intégrée à l'ordinateur.

REMARQUES Dans le premier format, en tant qu'instruction, la commande DATES désigne la valeur de l'horloge temps réel. MM est un nombre de deux chiffres entre 01 et 12 indiquant le mois. DD est un nombre de deux chiffres entre 01 et 31 indiquant le jour.

Dans le second format, la commande DATES renvoie la date de l'horloge temps réel dans un format MM/DD au programme comme variable alphanumérique. La variable jour avance d'un jour lorsque l'heure interne maintenu dans la variable TIMES passe de 23:59:59 à 00:00:00.

EXEMPLE:

```
>DATE$="12/25
```

Spécifie la date de l'horloge en temps réel au 25 décembre.

```
450:PRINT "AUJOURD'HUI NOUS SOMMES LE";DATK$
```

(4501 Imprime le jour actuel de l'horloge temps réel.

DEG

FORMAT: 1. DEG <dd.mmssrr>

Abréviation:

Voir également: DM S

BUT: Pour un angle spécifié en degrés, minutes et secondes, la commande DEG convertit la valeur en degré décimal.

REMARQUES L'angle doit être exprimé dans le format dd.mmssrr où dd représente le degré (gamme non limitée), mm la minute dans une gamme de 00 à 59, ss la seconde dans une gamme de 00 à 59 et rr le reste dans une gamme de 00 à 99. Notez que la partie en degrés (dd) est séparée d'avec le reste par un point décimal. DEG renverra l'angle en degrés et les degrés décimaux à 10 chiffres significatifs.

EXEMPLE:

```
10:X=DEG 50.300000
20:PRINT X
30:END
>RUN

 50.5
>
```

FORMAT: 1. DEGREE

DEGREE

Abréviation: DE.

Voir également: RADIAN, GRAD

BUT: Met l'ordinateur en mode degrés.

REMARQUES: En mode degrés, DEG est affichée sur la ligne au sommet de l'écran. L'introduction des arguments de SIN, COS et TAN doit être en degrés. Les valeurs renvoyées par les fonctions ASN, ACS et ATN seront en degrés décimaux.

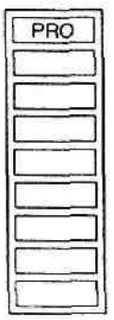
EXEMPLE:

```
10:DEGREE
20:PRINT "LES FONCTIONS TRIG SONT MAINTENANT EN DEGRES"
30:PRINT ASN(0.5),ASN(1.0)
40:PRINT ACS(0.5),ACS(0.1)
50:PRINT ATN(0.5),ATN(0.1)
60:END
```

Essaye de lancer ce programme et compare les résultats avec ceux de GRAD et RADIAN

[10] Choisit le mode DEGREE.
[30—50] Imprime certaines valeurs d'essai.

DELETE



FORMAT: 1. DELETE <numéro de ligne>
2.DELETE <numéro de ligne>
3.DELETE <numéro de ligne>, <numéro de ligne>
4.DELETE ,<numéro de ligne>

Abréviation: DEL. *Voir également:* NEW

BUT: Supprime les lignes de programme spécifiées dans la mémoire.

REMARQUES: La commande DELETE <numéro de ligne> supprime seulement la ligne de programme spécifiée. DELETE <numéro de ligne>, supprime les lignes déprogramme depuis le numéro de ligne spécifié jusqu'au numéro de ligne de programme le moins jeune dans la mémoire. DELETE <numéro de ligne>, <numéro de ligne> supprime toutes les lignes de programme entre le premier numéro de ligne spécifié (valeur inférieure) et le second numéro de ligne spécifié (valeur supérieure). DELETE ,<numéro de ligne> supprime les lignes de programme depuis le numéro de ligne le plus jeune dans la mémoire jusqu'au numéro de ligne spécifié.

Pour supprimer le programme entier, utiliser la commande NEW.

EXEMPLE:

```
>DELETE 150
```

Supprime la ligne 150 seulement.

```
>DELETE 200,
```

Supprime depuis la ligne 200 jusqu'au numéro de ligne le moins jeune.

```
>DELETE 50,150
```

Supprime toutes les lignes entre la ligne 50 et la ligne 150.

```
>DELETE ,35
```

Supprime depuis le numéro de ligne le plus jeune jusqu'à la ligne 35.

DIM

FORMAT: 1. DIM nom de variable numérique (<taille>)
2.DIM nom de variable alphanumérique (<taille>)[* <longueur>]
3.DIM nom de variable de tableau numérique (<rangées>,<colonnes>)
4.DIM nom de variable de tableau alphanumérique (<rangées>,<colonnes>) [* <longueur>]

Abréviation: D.
Voir également: CLEAR, ERASE

BUT: Utilisée pour réserver la place en mémoire pour les variables numériques, alphanumériques, de tableaux numérique et alphanumérique et pour spécifier le nombre d'indices inférieurs dans chaque tableau.

REMARQUES: Sauf les variables de valeur simple A-Z, A$-Z$, @(1)-@(26), et @$(1)-@$(26), tous les variables doivent être déclarées par une instruction DIM avant utilisation dans un programme, pour attribuer une place de stockage suffisante.

Le nom de variable numérique DIM (<taille>) spécifie le nombre d'éléments dans une variable numérique à une dimension dans une gamme de 0 à 255.

Le nom de variable de chaîne DIM (<faille>) [* <longueur>] spécifie le nombre d'éléments dans une variable alphanumérique à une dimension dans une gamme de 0 à 255. Le nombre maximum de caractères dans chaque élément est spécifié par le paramètre optionnel* <longueur> dans une gamme de 1 à 80. Si la longueur n'est pas spécifiée, la valeur par défaut est 16 caractères.

DIM nom de variable de tableau numérique (<rangées>, <colonnes>) spécifie le nombre de rangées et de colonnes dans un tableau numérique à deux dimensions dans une gamme de 0 à 255.

Le nom de variable de tableau alphanumérique DIM (<rangées>, <colonnes>) [* <longueur>] spécifie le nombre de rangées et de colonnes

dans un tableau alphanumérique à deux dimensions dans une gamme de 0 à 255. Le paramètre optionnel * <longueur> spécifie le nombre maximum de caractères dans chaque élément dans une gamme de 0 à 80.

La valeur minimum d'un indice inférieur quelconque est zéro, cela signifie que le nombre de rangées ou de colonnes dans un tableau est supérieur d'un à celui dans l'instruction DIM. Ainsi, par exemple, DIM A (2,3) déclare un tableau avec trois rangées et 4 colonnes:

	Col. 1	Col. 2	Col. 3	Col. 4
Rangéel	A(0,0)	A(0,1)	A(0,2)	A(0,3)
Rangée2	A(1,0)	A(1,1)	A(1,2)	A(1,3)
Rangée3	A(2,0)	A(2,1)	A(2,2)	A(2,3)

Le nombre total d'éléments est de 12.

Une fois un tableau dimensionné, il ne peut être redimensionné jusqu'à ce que cette taille soit réinitialisée par l'instruction CLEAR, NEW ou RUN ou sélectivement réinitialisée par l'instruction ERASE. Initialement, les éléments dans un tableau alphanumérique sont désignés à des chaînes vides.

Des codes d'erreur sont engendrés lorsqu'un programme cite un tableau non déclaré dans une instruction DIM, qu'une instruction DIM déclare un tableau déjà déclaré dans une instruction DIM précédente, et qu'un indice inférieur excédant la valeur spécifiée dans l'instruction DIM est cité.

EXEMPLE:

```
10:DIM C(13)

20:DIM F$(10)

30:DIM H(4,6)

40:DIM G$(7,5)*25
```

[10] Spécifie une variable numérique C avec 14 éléments.
[20] Spécifie une variable alphanumérique F$ avec 11 éléments.
[30] Spécifie un tableau numérique (5 rangées x 7 colonnes) de 35 éléments.
[40] Spécifie un tableau alphanumérique (8 rangées x 6 colonnes) de 48 éléments; chacun d'une longueur de 25 caractères.

FORMAT: 1. DMS <angle>

Abréviation: DM. *Voir également:* DEG

DMS

BUT: Convertit un angle spécifié en degrés décimaux en valeur en degrés, minutes et secondes.

REMARQUES L'angle est renvoyé sous le format dd.mmssrr où dd représente les degrés (gamme non limitée), mm, les minutes dans une gamme de 00 à 59, ss, les secondes dans une gamme de 00 à 59 et rr, le reste dans une gamme de 00 à 9. Noter que la partie en degrés est séparée du, reste par une virgule décimale. Le paramètre <angle> doit être introduit en degrés sous forme de dd. décimaux avec une virgule décimale.

EXEMPLE:

```
10:X=DMS  50.5
20:PRINT  X
30:END

>RUN
                          50.3
>
```

DSKF

FORMAT: 1. DSKF "d:"

Abréviation: DS.
Voir également:

BUT: Renvoie la quantité d'espace libre sur l'unité spécifiée.

REMARQUES: Les noms d'unité suivants peuvent être spécifiés.

S1: Unité dans le logement d'extension 1 S2: Unité dans le logement d'extension 2 X:, Y: Unité de disque souple

La quantité d'espace libre sur l'unité disponible pour le stockage des fichiers ou programmes est renvoyée sous forme d'un nombre entier en octets.

EXEMPLE:

```
>DSKF"S1:"
```

Renvoie l'espace disponible en octets sur le module dans le logement d'extension 1 (par exemple, module de disque RAM).

FORMAT; 1. END

Abréviation: E.
Voir également: STOP

END

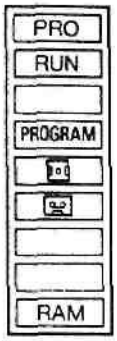
BUT: Arrête l'exécution du programme en cours, ferme tous les fichiers, et l'interface sérielle d'E/S si elle est ouverte.

REMARQUES: L'instruction END ne doit pas nécessairement se trouver sur la dernière ligne du programme. En cas de programmes contenant des sous-programmes en particulier, l'instruction END peut être placée avant les modules de sous-programme, de sorte que l'exécution ne continue pas dans le sous-programme lorsque la partie principale a été exécutée. Si END est omise, l'exécution se termine lorsqu'il ne reste aucune ligne de programme à exécuter, et les fichiers seront fermés.

EXEMPLE:

```
10:GOSUB 50
20:PRINT "APRES LE SOUS-PROGRAMME"
30:PRINT "LE PROGRAMME S'ARRETE ALA LIGNE 40"
40:END
50:PRINT "SOUS-PROGRAMME ALA LIGNE 50
60:RETURN
```

EOF



FORMAT: 1. EOF(<numéro de fichier>)

Abréviation: EO.
Voir également:

BUT: Renvoie une valeur qui indique quand la fin d'un fichier est atteinte en cours d'introduction depuis un fichier séquentiel.

REMARQUES: Le paramètre <numéro de fichier> est le numéro sous lequel le fichier a été ouvert par la commande OPEN.

La valeur renvoyée sera 0 si la fin du fichier n'est pas atteinte, ou 1 si la fin du fichier est atteinte. *EXEMPLE:*

```
10:IF EOF(1) THEN 100
```

[10| Fait un branchement sur la ligne 100 lorsque la fin du fichier n° 1 est atteinte.

ERASE

FORMAT: 1. ERASE <liste des noms de variables>

Abréviation: ERA. *Voir également:* CLEAR

BUT: Efface-les variables et tableaux spécifiés.

REMARQUES ERASE efface les variables numériques et alphanumériques et les tableaux, mais n'efface pas les variables fixes A-Z, A$-Z$, @(1) à @(26) et @$(1) à @$(26). Il n'est pas possible d'effacer les éléments individuels d'un tableau; l'ensemble de tableau est effacé. Un tableau est effacé en spécifiant le nom de tableau suivi de deux parenthèses (\.

EXEMPLE: Le paramètre <liste des noms de variables> spécifie les noms de variable, séparés par des virgules.

```
10:ERASE AB,Z$( )
```

ERL

FORMAT: 1. ERL

Abréviation:
Voir également: ERN, ON ERROR GOTO, RESUME

BUT: Renvoie le numéro de ligne auquel une erreur s'est produite en cours d'exécution.

REMARQUES: La fonction ERL est utilisée avec la fonction ERN et l'instruction ON ERROR GOTO dans les sous-programmes de traitement d'erreurs pour commander le déroulement du programme lorsqu'une erreur est produite. Un numéro de ligne est seulement désigné dans ERL si l'ecreur se produit en cours d'exécution du programme.

EXEMPLE: Voir ERN.

FORMAT: 1. ERN

Abréviation:

Voir également: ERL, ON ERROR GOTO, RESUME

ERN

BUT: Renvoie le numéro de code d'erreur de la dernière erreur d'exécution.

REMARQUES : La fonction ERN est utilisée avec la fonction ERL et l'instruction ON ERROR GOTO dans les sous-programmes de traitement d'erreurs pour commander le déroulement du programme lorsqu'une erreur est produite.

EXEMPLE:

```
10:ON ERROR GOTO 100
20:FOR N=1 TO 20
30:READ A
40:PRINT A
50:NEXT N
60:END
100:IF ERL =30AND ERN =4THEN PRINT "IL N'Y A PAS DE LIGNE DE D
110:STOP
```

EXP

FORMAT: 1. EXP(X)

Abréviation: EX. *Voir également:* LN

BUT: Renvoie la valeur de la fonction exponentielle "e" élevée à la puissance de X.

REMARQUES: L'expression X doit être comprise dans une gamme de +230,2585092 à -227,9559242. Pour les valeurs inférieures à cette gamme, 0 est renvoyé. Pour élever une autre base de numération à une puissance, utiliser la fonction "^". L'ordinateur maintient la valeur de "e" comme 2,718281828.

EXEMPLE:

```
>PRINT EXP(10)
                22026.46579
>
```

Imprime la valeur de "e" élevée à la puissance de 10.

FILES

FORMAT: 1. FILES "<d:>"
2.FILES "<d:nom de fichier>"
3.FILES "<d:nom de fichier ambigu>

Abréviation: FI.
Voir également: LFILES, SET

BUT: Affiche le répertoire des fichiers spécifiés sur le disque souple ou le disque RAM.

REMARQUES

La commande FILES affiche le nom de fichier, l'extension .BAS, la protection "P" (voir la commande SET) et les date et heure de création. Le paramètre d: spécifie l'unité.

1. Si aucun nom de fichier n'est spécifié, tous les fichiers dans l'unité spécifiée sont listés sur l'écran, une rubrique à la fois. Pour faire défiler l'image jusqu'à la rubrique suivante, appuyez sur la touche ↓. Pour suspendre le listage, appuyez sur n'importe quelle touche sauf les touches SHIFT, CTRL, DEF et SML.

2.Si un seul nom de fichier est spécifié, l'information de répertoire est affichée pour ce fichier.

3.Un nom de fichier ambigu peut être spécifié pour lister l'information de répertoire sur les groupes de fichiers ayant les formes de nom communes. Il existe deux caractères facultatifs à cet effet. L'astérisque "* " représente un nombre quelconque de caractères (y compris 0) dans le nom de fichier. Le point d'interrogation "?" représente un seul caractère dans le nom de fichier. Des exemples d'utilisation de ces caractères facultatifs sont indiqués ci-dessous.

Spécification du nom de fichier	Fichiers satisfaisant à la spécification
TEST?	TEST, TESTS, TEST1, TESTA
T??T	TEST, TEXT, TORT, TXYT
S7MPLE	SIMPLE, SAMPLE, S2MPLE, SOMPLE
A?????	ABCEDF, APPEND, APPLES, A12345
R*	RATES R1, RETURNS, RAND2, R, ROBERT

EXEMPLE:

```
>FILES"X:"
```

Liste tous les fichiers dans l'unité de disque souple X: sur l'écran.

```
>FILES"S1:DONNEE"
```

Affiche sur l'écran le fichier nommé DATA contenu dans le module de disque RAM dans le logement 1.

```
>FILES"S2:???1"
```

Liste sur l'écran tous les fichiers contenus dans le module de disque RAM dans le logement 2 qui ont les noms à 4 lettres se terminant par 1.

FOR.. NEXT

FORMAT: 1. FOR <compteur>=<valeur initiale> TO <valeur finale> [STEP <incrément>] NEXT <compteur> 2. FOR <compteur> = <expression1> TO <expression2> [STEP<expression3>] NEXT <comoteur>

Abréviation: F.~N. *Voir également:*

BUT: Permet l'exécution des lignes de programme entre FOR et NEXT à répéter un nombre spécifié de fois.

REMARQUES: L'instruction FOR..NEXT fait itérer l'exécution sur la même partie du programme à un nombre spécifié de reprises. Le paramètre <compteur> est un compteur qui maintient un enregistrement du nombre d'itérations qui ont été effectuées. Dans le format 1, les paramètres <valeur finale> et <incrément> sont des valeurs en nombres entiers compris dans une gamme de -32768 à +32767 (le paramètre <incrément> ne doit pas être zéro). Dans le format 2, ils peuvent se présenter sous forme d'expressions dont les valeurs servent de valeurs de compteur initiale et finale. Les combinaisons de ces deux formats sont admissibles. Les valeurs fractionnaires seront tronquées. Les lignes de programme qui suivent l'instruction FOR sont exécutées jusqu'à la rencontre de l'instruction NEXT, et le compteur est alors augmenté de la valeur spécifiée dans <incrément>, ou de 1 si aucun incrément n'est spécifié.

Ensuite, la valeur du compteur est comparée avec le paramètre <valeur finale>. Si la valeur du compteur est inférieure au paramètre <valeur finale>, l'exécution fait un branchement en amont sur l'instruction qui suit l'instruction FOR, et la séquence est répétée. Lorsque le paramètre <variable> est égal ou supérieur au paramètre <valeur finale>, l'exécution continue à partir de l'instruction qui suit l'instruction NEXT. Le paramètre <incrément> peut être une valeur numérique positive ou négative comprise dans une gamme de -32768 à +32767. S'il est négatif, le paramètre <va!eur finale> doit être inférieur au paramètre <valeur initiale>. Si le paramètre <valeur finale> est inférieur au paramètre <valeur initiale> pour un <incrément> positif, ou l'inverse pour un <incrément> négatif, la boucle

FOR..NEXT sera exécutée seulement une fois et l'exécution continuera à partir de la ligne qui suit l'instruction NEXT.

Les boucles FOR..NEXT peuvent être emboîtées; c'est-à-dire, une boucle FOR..NEXT peut être contenue dans une autre boucle. Ceci, en se combinant avec le second format de l'instruction qui utilise les expressions pour la spécification du compteur, peut rendre l'instruction FOR..NEXT la plus importante et la plus souple de toutes les commandes BASIC. Les instructions FOR-..NEXT doivent être appariées. Si une instruction NEXT est rencontrée avant l'instruction FOR correspondante, ou que les boucles sont emboîtées au-delà de la capacité de l'ordinateur, un code d'erreur sera produit.

EXEMPLE:

```
5:WAIT 30
10:FOR I=1 TO 5
20:PRINT I
30:NEXT I

40:FOR N=10 TO 0 STEP-1
50:PRINT N
60:NEXT N

70:FOR N=1 TO 10
80:LET X=1
90:FOR F=1 TO N
100:LET X=X*F
110:NEXT F
120:PRINT X
130:NEXT N
```

[5] Spécifie le temps d'attente pour la sortie sur papier de l'affichage. [10] Imprime les nombres 1, 2, 3, 4, 5.
[40' Décompte de 10 à 0 pour échelons de 1.
[70] Calcule et imprime la valeur de N factoriel par des nombres de 1 à 10.

GCURSOR

FORMAT: 1. GCURSOR X [,Y]

Abréviation: GC. *Voir également:*

BUT: Positionne le curseur graphique au point (X,Y) sur l'écran

REMARQUES: En mode graphique, l'écran est traité comme une matrice de 156 x 32 points. Le curseur graphique peut être positionné à n'importe quel point compris dans ces valeurs de coordonnées, ou même à l'extérieur de ces valeurs quoique, dans ce cas, le résultat ne soit pas visible, Après le positionnement du curseur graphique, l'instruction GPRINT peut être utilisée pour la sortie à cet emplacement sur l'écran.

La coordonnée X est normalement comprise dans une gamme de 0 à 155, et la coordonnée Y, dans une gamme de 0 à 31. Les deux coordonnées doivent, cependant, être comprises dans une gamme de -32768 à 32767. Si la coordonnée Y est omise, la coordonnée Y actuellement sélectionnée sera supposée.

NOTE:
NOTE:
En MODE 1 (mode PC-1500), le paramètre Y n'a pas de sens et ne doit pas être spécifié.

EXEMPLE:

```
:
:
300:X=5
310:Y=20
320:GOSUB 600
:
:
:
590:END
600:GCURSOR X,Y
610:GPRINT 255;255;255;255
620:RETURN
```

GCURSOR est utile pour le positionnement du curseur sur l'écran avant de dessiner le symbole ou la configuration spécifiés.

[300-320] Fixe l'emplacement sur l'écran du symbole à dessiner et appelle le sous-programme pour dessiner le symbole.
[600] Positionne le curseur aux valeurs XY spécifiées dans le corps principal du programme.
[610] Imprime le symbole. Ici, le symbole est un seul bloc de caractère d'une hauteur de 8 points et d'une largeur de 4 points.

GLCURSOR

FORMAT: 1. GLCURSOR (X,Y)

Abréviation: GL.
Voir également: LCURSOR

BUT: Déplace, en mode graphique, le stylo de l'imprimante à la position spécifiée.

REMARQUES : La commande GLCURSOR est similaire à la commande LCURSOR, mais déplace le stylo aux valeurs de coordonnées de graphique données sous forme de X, Y où X et Y sont des distances horizontale et verticale à partir de l'origine en vigueur comprises dans une gamme de -2048 à 2047. Le stylo est soulevé durant le mouvement.

EXEMPLE : Voir LLINE.

GOSUB..RETURN

FORMAT: 1. GOSUB <numéro de ligne/etiquette>

RETURN

Abréviation: G OS. RE.
Voir également: GOTO, ON..GOSUB, ON..GOTO

BUT: Les instructions GOSUB et RETURN sont utilisées pour sauter aux sous-programmes BASIC ou pour en revenir.

REMARQUES: Un sous-programme est un groupe de lignes de programme ou un 'module' de programme conçu pour remplir une fonction spécifique de façon itérative, à des moments différents dans le programme. Au lieu de répéter ces lignes chaque fois que la fonction est nécessaire, l'instruction GOSUB saute à la ligne de départ du sous-programme spécifié par le numéro de ligne ou l'étiquette. Elle se présente typiquement à la fin du programme principal, après l'instruction END, où elle ne sera pas rencontrée en cours d'exécution normale. La dernière instruction dans le sous-programme doit être une instruction RETURN. Les instructions dans le sous-programme sont exécutées et lorsque l'instruction RETURN est rencontrée, l'exécution fait un branchement amont sur la ligne dans le programme principal qui suit la ligne GOSUB. Le même sous-programme peut-être appelé à n'importe quel nombre de reprises à partir des GOSUB différentes, et un sous-programme peut être appelé par un autre.

EXEMPLE:

```
5:WAIT 150
10:GOSUB 70
20:PRINT "RENVOYE DU SOUS-PROGRAMME PRINCIPAL"
30:GOSUB 70
40:PRINT "RENVOYE POUR LA SECONDE FOIS"
50:END
60:REM APRES LE PROGRAMME PRINCIPAL, LE SOUS-PROGRAMME COMMENCE
   ICI
70:PRINT "SOUS-PROGRAMME PRINCIPAL SUR LA LIGNE 70"
80:GOSUB 100
90:RETURN
100:PRINT "SOUS-PROGRAMME EMBOITE ALA LIGNE 100"
110:PRINT "RETOUR AU SOUS-PROGRAMME PRINCIPAL"
120:RETURN

>RUN
"SOUS-PROGRAMME PRINCIPAL SUR LA LIGNE 70"
"SOUS-PROGRAMME EMBOITE ALA LIGNE 100
"RETOUR AU SOUS-PROGRAMME PRINCIPAL"
"RENVOYE DU SOUS-PROGRAMME PRINCIPAL"
"SOUS-PROGRAMME PRINCIPAL SUR LA LIGNE 70"
"SOUS-PROGRAMME EMBOITE ALA LIGNE 100"
"RETOUR AU SOUS-PROGRAMME PRINCIPAL"
>
```

[5] Spécifie le temps d'attente pour la sortie sur papier de l'affichage. [101 Appelle le sous-programme principal pour la première fois. [30] Appelle le sous-programme principal pour la seconde fois. [70-90] Sous-programme principal appelé par le programme principal [100-120] Sous-programme emboîté appelé par le sous-programme principal

FORMAT: 1. GOTO <numéro de ligne/étiquette>

GOTO

Abréviation: G.
Voir également: GOSUB..RETURN, ON..GOTO

BUT: Transfère inconditionnellement l'exécution au numéro de ligne spécifié.

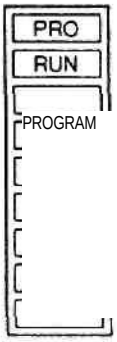
REMARQUES L'instruction GOTO est utilisée pour faire un branchement inconditionnel d'un point dans un programme à un autre. A la différence de l'instruction GOSUB..RETURN, l'instruction GOTO ne 'se rappelle pas' l'emplacement auquel l'exécution a fait un branchement. Si l'instruction sur la ligne spécifiée n'est pas exécutable (c'est-à-dire, une instruction DATA ou une ligne REM), l'exécution commence par la première ligne exécutable après celle-ci. Si une ligne non existante est spécifiée, un code d'erreur se produit. GOTO peut également être utilisée pour continuer l'exécution après l'apparition d'une BREAK (voir CONT). GOTO seule fait commencer l'exécution à partir de la première ligne dans le programme.

EXEMPLE:

```
10:INPUT A$
20:IF A$="O"THEN 50
30:PRINT "NON"
40:GOTO 60
50:PRINT "Oui"
60:END
```

Imprime OUI si un O est introduit par clavier, et imprime NON si n'importe quoi d'autre est introduit.

GPRINT



Abréviation: G P.
Voir également: G C U R S 0 R

FORMAT: 1. GPRINT[SET/OR/XOR,]<vaieur d'image binaire>;
<valeur d'image binaire>;.....
2.GPRINT[SET/OR/XOR,]"<chaîne d'image binaire hexadécimale>"
3.GPRINT

BUT: Trace les configurations graphiques sur l'écran.

REMARQUES: GPRINT affiche sur l'écran une colonne ou une série de colonnes des données d'image binaire à partir de la position actuelle du curseur graphique. Les données d'image binaire peuvent être spécifiées de trois manière; sous forme d'une liste des valeurs décimales ou des valeurs hexadécimales séparées par des points-virgules, ou sous forme d'une seule chaîne hexadécimale. De chacune de ces manières, chaque article de données est converti en nombre binaire à 8 bits équivalent, et chacun des huit bits est utilisé pour mettre un point en circuit (noir) ou hors circuit (clair). Donc, un article de données d'image binaire désigne sur l'écran les huit points dans une seule colonne dont la hauteur est égale à une ligne de caractères. Une liste ou chaîne des articles de données d'image binaire désignera les points le long d'une ligne sur l'écran ayant la même hauteur qu'une ligne en mode de texte:

Exemple 1 :

x – bit 1 = 1
x – bit 2 = 1
x – bit 3 = 1
x – bit 4 = 1
x – bit 5 = 1
x – bit 6 = 1
x – bit 7 = 1
x – bit 8 = 1

bit 8 bit 1
↓ ↓
Donnée d'image binaire = 11111111 en binaire
= 255 en décimal
= FF en hexadécimal

Exemple 2:

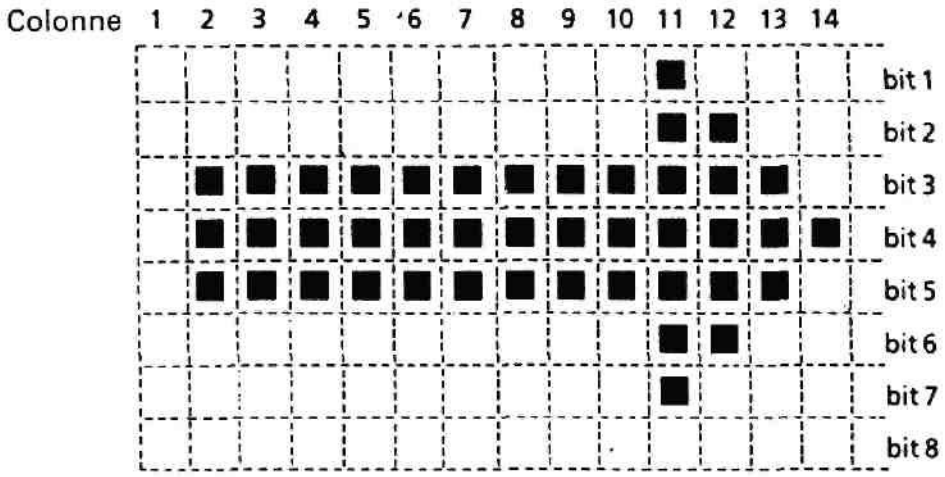
– bit 1 = 0
x – bit 2 = 1

bit 8 bit 1
↓ ↓
Donnée d'image binaire = 11010110 en binaire
= 214 en décimal
= D6 en hexadécimal

X - bit 3 =* 1
- bit 4 = 0
X - bit 5 = 1
- bit 6 = 0
X - bit 7 = 1
X - bit 8 = 1

Si la valeur d'article de données d'image binaire est donnée en hexadécimal, elle doit être précédée par un signe "&" (c'est-à-dire, &D6, &FF) sauf lorsqu'elle est spécifiée sous forme d'une chaîne hexadécimale de format 3. Sous cette forme, la gamme est de &00 à &FF. Si la chaîne contient un nombre impair de caractères, le dernier caractère de la chaîne sera ignoré.

Exemple 3:



La colonne 1 ne contient aucun point, et la donnée est donc
00000000 = &0.
La donnée d'image binaire est 00011100 pour les colonnes
2 à 10 = &1C
01111111 pour la colonne 11 = &7F
00111110 pour la colonne 12 = &3E
00011100 pour la colonne 13 = &1C
et 00001000 pour la colonne 14 = &08
Per conséquent, l'instruction suivante imprimera une grande flèche graphique sur l'écran:

GPRINT&0; &1C; &1C; &1C; &1C; &1C; &1C; &1C; &1C; &1C;&7F;&3E;&1C;&08

Les instructions suivantes spécifient les mêmes configurations de points pour les formats 1 et 2:

GPRINT 16;40;18;253;18;40;16 (Format 1, décimal)
GPRINT &10;&28;&12;&FD;&12;&28;&10 (Format 1, hexadécimal)
GPRINT "102812FD122810" (Format 2, chaîne hexadécimale)

Les paramètres SET, OR et XOR sont spécifiés de la façon suivante:

SET — La configuration de points est désignée tel que spécifié dans l'image binaire.
OR — Les désignations des points spécifiés dans l'image binaire seront soumises à l'opération OR avec les désignations de points existants à cet emplacement sur l'écran (voir la description de l'opérateur OR).
XOR — La désignation de point spécifié dans l'image binaire sera soumise à l'opération OR exclusive avec les désignations des rJoints existantes à cet emplacement sur l'écran.

L'option SET est la commande par défaut.

GRPINT seule déplacera le curseur graphique d'une ligne vers le bas sans effacer aucun affichage existant sur cette ligne.

EXEMPLE:

```
10:CLS
20:FOR I=1 TO 77
30:GPRINT 255;214;
40:NEXT I
50:END
```

[10J Efface l'écran.
[20] GPRINT définit deux valeurs d'image à 8 bits et chaque fois que la ligne 30 est exécutée, une zone de 8 x 2 (hauteur x largeur) est affichée sur l'écran. La largeur d'écran étant de 156 bits ou points, l'exécution de ce programme remplira une seule ligne sur l'écran sauf pour les deux dernières colonnes à 8 bits. Noter que la ligne 20 se termine en un point-virgule. Ceci est important pour empêcher les retours chariots d'être ajoutés après l'exécution de chaque GPRINT qui réalise l'impression de configuration sur 77 lignes différentes.
[30] Les mêmes configurations que celles décrites ci-dessus dans les exemples de commande 1 et 2 sont répétées 77 fois.

FORMAT: 1. GRAD

GRAD

Abréviation: GR.
Voir également: DEGREE, RADIAN

BUT: Met l'ordinateur en mode gradients.

REMARQUES En mode gradients, GRAD s'affiche à la ligne la plus haute de l'écran. Les arguments de SIN, COS et TAN doivent être introduits sous forme d'une valeur de gradient. Les valeurs renvoyées par les fonctions ASN, ACS et ATN seront des valeurs de gradients.

EXEMPLE:

```
10:GRAD
20:PRINT "LES FONCTIONS TRIG SONT MAINTENANT EN VALEURS DE
   GRADAIENTS"
30:PRINT ASN(0.5),ASN(1.0)
40:PRINT ACS(0.5),ACS(1.0)
50:PRINT ATN(0.5),ATN(1.0)
60:END
```

Exécutez ce programme et comparez les résultats avec ceux pour DEGREE et RADIAN.

[10] Sélectionne le mode GRAD.
[30-50] Imprime quelques valeurs d'exemple.

GRAPH

FORMAT: 1. GRAPH

Abréviation: GRAP. *Voir également:* TEXT

BUT: Met l'ordinateur en mode graphique.

REMARQUES: Lorsque GRAPH est exécutée, les paramètres de gamme d'impression de la commande PAPER <limite en sens inverse> et <limite en sens direct> sont remis à leurs valeurs par défaut.

EXEMPLE:

```
>GRAPH
```

L'ordinateur sélectionne le mode GRAPH pour l'imprimante.

HEX$

FORMAT: 1. HEX$(<nombre décimal>)
2. HEX$(&<nombre hexadécimal)

Abréviation: H. *Voir également:* VAL

BUT; Renvoie une chaîne de caractères représentant une valeur hexa décimale.

REMARQUES : Le <nombre décimal> spécifié doit être compris dans la gamme allant de 0 à 65535. La chaîne de caractères renvoyée sera comprise dans la gamme allant de &0 à &FFFF. Les nombres non entiers seront arrondis au nombre le plus proche avant le renvoi de la chaîne. Pour convertir un nombre hexadécimal en sa chaîne équivalente, utiliser "&" avant la valeur.

EXEMPLE:

```
10:PRINT "CONVERSION DECIMALE EN HEXADECIMALE"
20:INPUT "INTRODUIRE LE NOMBRE DECIMAL",X
30:IF X>65535THEN 100
40:IF X<0THEN 110
50:D$=HEX$(X)
60:PRINT "HEX EQUIVALENT DE";X;" EST ";D$
70:INPUT "UN AUTRE? TAPEZ O OU N ",A$
80:IF A$="O" THEN 20
90:GOTO 120
100:PRINT "LE NOMBRE EST TROP GRAND":GOTO 20
110:PRINT "LE NOMBRE DOIT ETRE BE+VE":GOTO 20
120:END
```

IF.THEIM

FORMAT: 1. IF<condition>THEN<numéro de ligne/étiquette/instruction> 2. IF<condition>THEN<numéro de ligne/étiquette/instruction> ELSE<numéro de ligne/étiquette/instruction>

Abréviation: IF T. ~ EL.

-Voir également:

BUT: Change l'ordre d'exécution du programme, suivant les résultats de la condition spécifiée au moment où le programme est exécuté.

REMARQUES La paire IF..THEN permet de prendre les décisions dans un programme BASIC. La clause THEN suivant la condition IF est exécutée si le résultat de la condition est vrai. S'il est faux, la clause THEN est ignorée et la clause ELSE est exécutée. S'il n'existe aucune clause ELSE, l'exécution continue à partir de la ligne suivante. Le paramètre <condition> peut être une égalité simple telle que X = 1 ou une expression logique plus complexe. La clause THEN ou ELSE peut spécifier n'importe quelle instruction BASIC (ordinairement une instruction d'affectation ou de branchement GOTO).

Les instructions IF..THEN..ELSE peuvent être emboîtées sur une ligne de programme jusqu'à la longueur maximum de ligne de programme (80 caractères).

EXEMPLE:

```
10:INPUT "A CONTINUER? ",A$
20:IF A$="OUI" THEN 10
30:IF A$="NON" THEN 60
40:PRINT "PRIERE D'INTRODUIRE OUI OU NON"
50:GOTO 10
60:END
```

Le programme persiste à demander "A CONTINUER?" tant que OUI est introduit; il s'arrête si NON est introduite et réclame autrement.

INIT TM

FORMAT: 1. INIT "Sn:" , "F"
"M"
"P"

2.INIT"X:"

3.INIT "COMn:",<capacité du tampon>

Abréviation: INI.

Voir également: SETCOM, SETDEV, TITLE

BUT:

1. Initialise les modules dans les logements d'extension 1 ou 2 et spécifié l'usage.
2. Initialise et prépare un disque souple vierge.
3. Désigne la capacité du tampon de réception pour les deux ports série.

REMARQUES: Format 1 :

Sn: initialise le module de disque RAM dans le logement d'extension S1: (logement 1) ou S2: (logement 2).

Ce format n'est effectif qu'en MODE 0.

Le module dans le logement d'extension spécifié peut être initialise comme un disque RAM, une mémoire de programme ou une extension de mémoire de la façon suivante:

Paramètre	Usage
"F"	Prépare le module comme un disque RAM. Ceci permet aux programmes et fichiers d'être mémorisés de la même manière que sur le disque souple. Voir Chapitre 11-Fichiers, pour plus de détails sur la mémorisation des fichiers sur le disque RAM.
"M"	Initialise le module comme une extension de la zone utilisateur intérieure de PC-1600. Ceci permet l'exécution des programmes d'encombrement plus important.
P"	Initialise le module comme une zone de stockage de programme. Un seul programme peut être écrit sur le module et sera conservé par là batterie de secours même lorsque le module est extrait de l'ordinateur.

La commande INIT ne peut pas être exécutée sur un module qui contient déjà des programmes ou fichiers, ou lorsque l'interrupteur d'interdiction d'écriture est mis en position ON.

Format 2:
INIT "X:" initialise et prépare un disque souple vierge. Tous tes nouveaux disques doivent être préparés par cette commande avant leur utilisation. Si un disque dispose déjà de quelque information mémorisée, l'exécution de la commande INIT effacera effectivement tous les contenus.

Format 3:
La capacité du tampon de réception pour un port série peut être désignée dans une gamme de 80 à 16383 octets, ou à une valeur minimum de 40 octets en spécifiant 0 comme capacité du tampon. Les ports suivants peuvent être spécifiés par COMn:

COM1 : Port série RS-232C
COM2: Port série optique
COM: Port actuellement sélectionné par l'instruction SETDEV.

La valeur de <capacité du tampon> est remise à 0 (40 octets) lors de la mise sous tension ou après une opération de remise à zéro.

Si les fichiers sont en cours d'échange entre deux ordinateurs qui sont câblés ensemble (c'est-à-dire, sans impliquer aucun modem ni ligne téléphonique), un grand tampon de réception accélérera le transfert de fichier. D'autre part, si les fonctions de communication ne sont pas utilisées, la désignation de la capacité du tampon à la valeur minimum (0 = 40 octets) libérera l'espace mémoire pour d'autres buts. La moyenne type de capacité du tampon pour la désignation est de 256 octets.

S'il existe une mémoire insuffisante disponible pour la capacité du tampon spécifiée ou qu'un fichier est actuellement ouvert sous l'option APPEND de l'instruction OPEN, un code d'erreur se produit.

EXEMPLE:

```
>INIT"S1:",,"F"
```

Initialise le module dans le logement 1 comme un module de disque RAM.

```
>INIT"X:"
```

Initialise le disque souple dans l'unité de disque souple en te préparant. Toute donnée existante sur le disque souple est perdue.

INKEY$

FORMAT: 1. <variable alphanumérique>-INKEY$
2.<variab!e alphanumérique>=INKEY$(0)
3.<variable alphanumérique>=INKEY$(1)

Abréviation: INK.
Voir également:

BUT: Contrôle le tampon du clavier durant l'exécution du programme, et introduit un seul code de touche dans le programme.

REMARQUES: INKEY$ et INKEY$(0) ont le même sens. INKEY$ ou INKEY$(0) lit le dernier caractère de touche par le tampon de clavier et l'affecte au paramètre <variab|e de chaîne>. Si aucune touche n'est enfoncée, INKEY$ renvoie une chaîne vide (code 0 d'ASCII). Si le tampon du clavier contient plusieurs introductions au clavier, INKEY$(1) peut être utilisée pour lire à partir de celle la plus ancienne. Le caractère lu depuis le tampon du clavier n'est pas affiché comme un caractère sur l'écran.

Seuls les caractères listés dans la table peuvent être lus en utilisant la commande INKEY$:

Table de caractères INKEY$

Bas \ Haut	0	1	2	3	4	5	6
0			SPACE	0		P	
1	SHIFT	F1		1	A	Q	
2	SML	F2		2	B	R	
3	CTRL	F3		3	C	S	
4	KBD/CLICK	F4		4	D	T	
5	BS	F5		5	E	U	
6		F6		6	F	V	
7				7	G	W	
8	◀	CL	(	8	H	X	
9	⇕	RCL	)	9	I	Y	
A	↓		*		J	Z	
B	↑	DEF	+		K		
C	▶				L		
D	ENTER		–	=	M		
E	ON		.		N		
F	OFF	MODE	/		O		

EXEMPLE:

```
300:A$=INKEY$
310:IF A$="" THEN 300
320:IF A$="0"THEN 500
330:GOTO 300
:
:
500:PRINT "OUI....."
```

Cette partie du programme est similaire à l'utilisation de l'instruction INPUT. Le programme attend l'enfoncement d'une touche par l'utilisateur. L'instruction INKEY$ diffère de l'instruction INPUT par le fait qu'elle n'utilise pas l'écran pour l'introduction. Le programme maintient l'itération jusqu'à ce qu'une touche soit enfoncée.

[300] Lit un caractère du tampon de clavier et l'affecte à la variable de chaîne A$.
[310] Maintient le contrôle du tampon de clavier jusqu'à l'enfoncement d'une touche.
[320] Fait un branchement sur la ligne 500 si "0" est introduit.
[330] Fait un branchement amont sur la ligne 300 si un caractère quelconque sauf "0" est introduit.

INP

FORMAT: 1. INP(<adresse d'accès>)

Abréviation:

Voir également: OUT

BUT: Renvoie une valeur correspondant à l'entrée du port du micro processeur Z-80A directement.

REMARQUES: INP renvoie la valeur d'une configuration binaire introduite directement à un port d'entrée de machine.

Le paramètre <adresse d'accès> spécifie le port d'entrée de machine sous forme d'une adresse à 16 bits comprise dans une gamme de 0 à 65535 (&0 à &FFFF) et renvoie un octet de données à partir de l'adresse du port.

EXEMPLE:

```
:
:
300:A=INP (20)
:
:
```

Renvoie l'entrée de valeur à l'adresse du port de machine 20 (&14).

INPUT

FORMAT: 1. INPUT["<chaîne de message>"(; l]<!iste des variables>

2. INPUT <chaîne de message>" ; l<vanable>,"<chaîne de message>"j ; l<variable>,...

Abréviation: I.
Voir également:

BUT: Permet aux valeurs d'être introduites par l'intermédiaire du clavier dans les variables du programme.

REMARQUES Lorsque INPUT est rencontrée, l'exécution s'arrête et affiche la chaîne de message éventuelle. Les données peuvent alors être introduites par le clavier ou à travers un port série E/S si ainsi désigné par SETDEV, et seront affectées au(x) nom(s) de variable spécifié(s) dans la <liste des variables>. Les variables dans la <liste des variables> doivent être séparées par des virgules. Les valeurs introduites par clavier doivent être introduites une à une en les séparant par ENTER.

S'il n'existe aucune chaîne de message, un point d'interrogation est affiché pour indiquer à l'utilisateur d'introduire les données. Si la chaîne de message est suivie par un point-virgule, aucun point d'interrogation n'est affiché, et le curseur est positionné après le dernier caractère de la chaîne de message. Si la chaîne de message est suivie par une virgule, le curseur se déplace à la position avant de la ligne au-dessous de la chaîne de message et attend l'introduction de données.

Les données peuvent également être introduites à partir de l'accès RS-232C par la commande INPUT sous le format suivant:

INPUT<variable>[,<variable>] ....

Aucune chaîne de message ne peut être spécifiée et aucun point d'interrogation n'est affiché.

EXEMPLE:

```
10IPRINT "VOLUME DE SOLIDE"
```

```
20:iNPUT "INTRODUIRE L.B.H "L.B.H 3
0:V = L * B*H
40:PRINT "LE VOLUME EST"IV
5O:END

>RUN
VOLUME DE SOLIDE
I NTRODU I RE L.B.H
3ENTER
4ENTRE
5ENTER
LE VOLUME EST 60 >
```

[20] Demande à l'utilisateur d'introduire trois valeurs numériques. Le message ci-dessus apparaît sur l'écran lorsque le programme est exécuté. Dans l'exemple de RUN, l'utilisateur introduit 3, 4 et 5 comme valeurs.

[30-40] Calcule le volume et imprime la réponse.

INPUT#

FORMAT: 1. INPUT#<numéro de fichier>,<liste des variables> 2.
INPUT#["<nom de fichier>";]<liste des variables>

: l.# *Voir également:* DIM, INPUT, OPEN, PRINT#, SET

BUT: Utilisée pour lire les articles d'un fichier séquentiel sur le disque souple, le module de disque RAM ou la bande cassette.

REMARQUES: 1. Disque ou module de disque RAM
Le paramètre <numéro de fichier> est le numéro affecté au fichier lorsqu'il a été ouvert pour l'introduction par l'instruction OPEN. Une tentative d'introduction des articles d'un fichier qui n'a pas été ouvert produira un code d'erreur.

2. Fichiers sur bande cassette
La valeur par défaut lorsque le "<nom de fichier>" n'est pas spécifié pour la bande cassette est le premier fichier à partir de la position actuelle de la tête de la bande.

Le paramètre <liste des variables> spécifie les noms des variables dans lesquelles les articles de données seront écrits lorsque l'instruction INPUT# est exécutée. La liste peut contenir les variables fixes, variables simples et variables composées délimitées par des virgules. L'ordre dans lequel les variables sont spécifiées dans la liste, doit correspondre à l'ordre dans lequel les données apparaissent dans le fichier; chaque type de variable doit avoir un type de données correspondant.

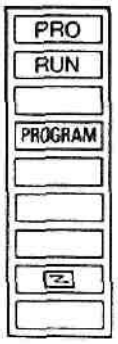
Les variables alphanumériques doivent également avoir une longueur suffisante pour enregistrer les articles de données. Si les tableaux sont enregistrées par l'instruction INPUT#, l'espace mémoire nécessaire doit être déclaré dans le programme par DIM.

Si le nombre d'articles de données enregistrés à partir du fichier est inférieur au nombre de variables dans la liste, l'exécution s'arrête en état d'attente. Terminer cet état en appuyant sur la touche BREAK. Si le nombre d'articles de données dans le fichier est supérieur au nombre de variables dans la liste, les articles de données en excès ne seront pas enregistrés.

En format 2 pour les fichiers sur bande cassette, les variables composées doivent être spécifiées sous la forme A(*) en utilisant un astérisque pour l'élément. Les éléments simples d'un tableau ne peuvent pas être spécifiés. Cette forme ne peut pas être utilisée dans le format 1.

EXEMPLE: Voir OPEN.

INSTAT



FORMAT: 1. INSTAT "COM1:

Abréviation: INSTA.

Voir également: OUTSTAT

BUT: Renvoie les désignations des signaux de commande pour le port série RS-232C.

REMARQUES: Les désignations des signaux de commande sont renvoyées sous forme d'un nombre binaire à 8 bits. Chaque bit représente la valeur d'un signal de commande de la façon suivante:

Bit	Valeur	Condition de signal
1	0	DTR est haut.
	1	DTR est bas.
2	0	RTS est haut.
	1	RTS est bas.
3	0	CTS est haut.
	1	CTS est bas.
4	0	CD est haut.
	1	CD est bas.
5	0	DSR est haut.
	1	DSR est-bas.
6	0	CI est haut. CI
	1	est bas.
7	toujours 0	
8	toujours 0	

EXEMPLE:

```
>INSTAT "COM1I"
```

Renvoie les désignations des signaux de commande pour le port série RS-232C. Ici, un nombre décimal de 63 correspond à un nombre hexadécimal de 3F et à la configuration binaire suivante:

Tous les six signaux sont alors spécifiés comme bas (le bit 1 est le bit de poids fort), soit la condition par défaut.

PRO
RUN

INSTR

PROGRAM

FORMAT: 1. INSTR([<colonne>,]X$,Y$)
2. INSTR([<colonne>,]"<chaîne>","<caractère>")

Abréviation: INS.
Voir également:

BUT: Explore la chaîne X$ pour chercher la première apparition du caractère Y$ et renvoie la position sous forme d'un nombre entier.

REMARQUES Si Y$ ne peut pas être trouvé ou que XS est une chaîne vide, la valeur 0 est renvoyée. Dans le format 2, le caractère réel est spécifié avec une chaîne, réelle, délimités tous les deux par des guillemets. Le paramètre <colonne> est la colonne de départ pour la recherche. La valeur par défaut est la colonne 1.

EXEMPLE:

```
10:INPUT "INTRODUIRE UN MOT ";A$
20:N=INSTR (A$,"A")
30:IF N=0 THEN 60
40:PRINT "LE PREMIER 'A' EST LA LETTRE #";N
50:GOTO 70
60:PRINT "PAS DE LETTRE 'A' DANS LE MOT"
70:END

>RUN
INTRODUIRE UNE DONNEE DE MOT
LA PREMIER LETTRE 'A' EST LA LETTRE#2
>
```

Ce programme cherche 'A' dans un mot.

[10] L'utilisateur introduit un mot quelconque par clavier. [201 L'instruction INSTR cherche tout 'A' dans le mot. [40-50] L'emplacement du premier 'A' est imprimé. [601 Aucun 'A' n'a été trouvé dans le mot.

FORMAT: 1. INT(X)

INT

Abréviation: Voir également:

BUT: Tronque la partie décimale d'un nombre réel et renvoie un nombre entier.

REMARQUES: La valeur X est arrondie au nombre entier inférieur. Le nombre entier obtenu est toujours égal ou inférieur à X indifférement à ce que X est positif ou négatif. Pour les valeurs négatives, le nombre entier absolu obtenu sera alors égal ou supérieur à la valeur absolue.

EXEMPLE:

```
5:WAIT 60
10:A=-3.3:PRINT A,INT(A)
20:B=-1.9:PRINT B,INT(B)
30:C=-0.5:PRINT C,INT(C)
40:D=0.2:PRINT D,INT(D)
50:E=1.6:PRINT E,INT(E)
60:F=3:PRINT F,INT(F)

>RUN
-3.3          -4
-1.9          -2
-0.5          -1
 0.2           0
 1.6           1
 3             3
>
```

KBUFF$

FORMAT: 1. KBUFF$=<chaîne de caractères>

Abréviation: KB.
Voir également: INKEY$

BUT: Introduit une chaîne de caractères dans le tampon de clavier.

REMARQUES : La commande KBUFFS peut être utilisée pour faire fonctionner l'ordinateur en mode "séquentiel". Les commandes du système peuvent être introduites dans un fichier des commandes dans l'ordre de leur exécution, et enregistrées dans le tampon de clavier par KBUFF$. Ceci permet aux commandes d'être exécutées comme si elle avait été introduites directement au clavier.

Les fichiers des commandes ou fichiers séquentiels peuvent être utilisés pour exécuter automatiquement une série de commandes pour une certaine tâche. Par exemple, un fichier des commandes peut être introduit pour remettre la mémoire à zéro, charger un programme spécifique, exécuter le programme et récrire les résultats dans un fichier.

La longueur de la chaîne de caractères peut être de 32 caractères au maximum. La chaîne recouvrera tout contenu existant du tampon.

EXEMPLE:

```
:
:
200:KBUFF$ ="45"
210:INPUT A
220:PRINT "A EST";A
:
:
```

[200] Introduit 45 dans le tampon de clavier comme si vous aviez introduit par clavier le nombre en cours d'exécution du programme.
[210] INPUT attend l'introduction d'un nombre par l'utilisateur, mais cette opération a été déjà effectuée "a l'aide de la commande KBUFF$. Notez qu'il n'existe aucun retour chariot introduit avec la chaîne KBUFF$. Lorsque le programme est exécuté, 45 apparaît automatiquement sur l'écran, mais vous devrez encore appuyer sur la touche d'introduction. Voir CHRS et les tables de code ASCII pour introduire dans les variables alphanumériques les caractères de touche qui ne s'impriment pas. (220) Imprime la valeur de A.

KEY ON/OFF/STOP

FORMAT: 1. KEY(<numéro de touche>)ON
2.KEY(<numéro de touche>)OFF
3.KËY(<numéro de touche>)STOP

Abréviation:
Voir également: ON KEY GOSUB

BUT: Valide ou invalide la touche de fonction spécifiée utilisée pour causer un branchement du programme durant l'exécution.

REMARQUES: Cette instruction commande l'action de l'instruction ON KEY GOSUB.

KEY(<numéro de touche>) OFF invalide l'une des touches de fonction F1 à F6 spécifiée par le <numéro de touche> compris dans une gamme de 1 à 6. L'enfoncement de la touche n'exercera aucun effet sur une instruction ON KEY GOSUB après l'exécution de ladite instruction. KEY (<numéro de touche>) ON valide une touche de fonction qui a été invalidée par l'instruction KEY(<numéro de touche>) OFF/STOP. Ces deux commandes peuvent être émises dans un programme afin d'éviter les interruptions du programme.

KEY(<numéro de touche>) STOP invalide une touche de fonction. Cependant, si une instruction KEY ON est exécutée plus tard lorsque la touche de fonction a été enfoncée durant la période où KEY OFF était effective, l'exécution fera imédiatement un branchement suivante l'instruction ON KEY GOSUB. La commande par défaut est KEY STOP.

Une touche qui est affectée dans une instruction ON KEY GOSUB ne peut pas servir de touche de fonction normale pour introduire les chaînes ou commandes prédéterminées.

EXEMPLE:

```
10:KEY(3) OFF
:
:
:
390:KEY(3) ON
```

[101 Invalide la touche de fonction F3 durant l'exécution du programme.
[390] Revalide la touche de fonction F3 pour l'utilisation normale.

KEYSTAT

FORMAT: 1. KEYSTAT, [< para m être 1>][,<paramètre 2>]

Abréviation: KE.
Voir également: IN KEY$

BUT: Pour commander l'introduction par clavier, spécifie l'unité d'entrée standard, active/désactive la fonction de répétition de touche et la fonction de cliquetis de touche.

REMARQUES: Le <paramètre 1> active/désactive la fonction de répétition:

0— répétition de touche désactivé

1— répétition de touche activée

Le <paramètre 2> met le cliquetis de touche en/hors action:

0— cliquetis de touche hors action
1— cliquetis de touche en action

Lors de la mise sous tension, la désignation de la valeur par défaut conserve toujours les dernières valeurs spécifiées. Après une ALL RESET, la désignation par défaut est KEYSTAT,0,0.

EXEMPLE:

```
>KEYSTAT,1,1
```

Active à la fois la fonction de répétition de touche et la fonction de cliquetis de touche.

KILL

FORMAT: 1. KILL "<d:nom de fichier>"

Abréviation: K.
Voir également: SAVE, SET

BUT: Utilisée pour supprimer un fichier dans le disque souple ou le module de disque RAM.

REMARQUES: Le paramètre <d:nom de fichier> spécifie l'unité et le nom de fichier. Dans le cas d'un fichier BASIC, l'exécution de .BAS doit être spécifiée avec le nom de fichier. Si le fichier est protégé par l'attribut "P" de l'instruction SET ou que l'interrupteur d'interdiction de l'écriture du module RAM est mis en circuit, un code d'erreur se produit. Un fichier qui est actuellement ouvert ne peut pas être supprimé jusqu'à ce qu'il soit fermé.

EXEMPLE:

```
>KILL"S1:DONNEE PRECEDENTE"
>
```

Efface le fichier OLDATA contenu dans le module de disque RAM dans le logement d'extension 1.

LCURSOR

FORMAT: 1. LCURSOR <colonne>

Abréviation: LC.
Voir également: GLCURSOR, PCONSOLE, TAB

BUT: Déplace, en mode de texte, le stylo d'imprimante jusqu'à la colonne spécifiée.

REMARQUES: La commande LCURSOR est similaire à la commande GLCUR-SOR, mais déplace le stylo jusqu'à la colonne d'impression spécifiée dans la gamme allant de 0 à la longueur maximum de ligne désignée par l'instruction PCONSOLE en cours d'exécution.

EXEMPLE:

```
10:LCURSOR 40
20:LPRINT "MILIEU"
30:LPRINT "GAUCHE"
40:END
```

Ce programme n'est exécuté qu'en mode TEXT.

[10] Déplace le chariot porte-stylo au milieu du papier.
120] Ce message fera commencer l'impression à partir de la position actuelle de l'imprimante au milieu de la page.
[30] Après la dernière LPRINT, la tête d'imprimante recule vers la gauche. Ainsi, cette ligne s'imprime en position normale cadrée à gauche.

LEFT$

FORMAT: 1. LEFT$(X$,N)
2. LEFT$("<chaîne>",N)

Abréviation: LE F.
Voir également: MID$, RIGHT$

BUT: Renvoie N caractères à partir de l'extrémité gauche d'une chaîne quelconque, X$.

REMARQUES : La valeur de N doit être comprise dans une gamme de 0 à 80. Les fractions seront arrondies à l'unité inférieure (tronquées). Si N est inférieur à 1, une chaîne vide est renvoyée. Si N est supérieur au nombre de caractères flans X$, la chaîne entière est renvoyée.

EXEMPLE:

(X$,N)

```
10:X$="SHARP"
20:FOR N=1 TO 6
3O:LET S$=LEFT$
40IPRINT S$ 5
OINEXT N

>RUN
S
SH
SHA
SHAR
SHARP
SHARP
>
```

LEN

FORMAT: 1. LEN(X$)
2. LEN("<chaîne>")

Abréviation: Voir également:

BUT: Renvoie le nombre de caractères dans la chaîne X$.

REMARQUES: Le nombre de caractères dans la chaîne comprend tout espace ou caractère qui ne s'imprime pas tel que le code de commande ou le retour chariot.

EXEMPLE:

```
10:INPUT "INTRODUIRE UN MOT ";A$
20:N=LEN A$
30:PRINT "LA LONGUEUR DU MOT EST:";N
40:END

>RUN
INTRODUIRE UN MOT CERISE
LA LONGUEUE DU MOT EST 6
>
```

[10] Introduit un mot. Dans cet exemple, l'utilisateur introduit "CERISE". [20] Trouve la longueur du mot. [30] Imprime la réponse.

LET

FORMAT: 1. [LET]<variable>=<expression>
[,<variable> = <expression>].....

Abréviation: LE.
Voir également:

BUT: Affecte une valeur a une variable.

REMARQUES: Le mot [LET] est optionnel sauf dans les instructions incluses dans une clause THEN ou ELSE de l'instruction IF.THEN..ELSE. Le type d'expression doit correspondre au type de variable. Seules les expressions numériques peuvent être affectées aux variables numériques, et; seules les expressions alphanumériques peuvent être affectées aux variables alphanumériques.

EXEMPLE:

```
10:LET X$= "PREMIERE CHAINE"
20:Y$= "DEUXIEME CHAINE"
30:PRINT X$:PRINT Y$
40:LET A=6
50:B=A+1
60:LET R$=LEFT$ (X$,A)
70:S$=LEFT$ (Y$,B)
80:PRINT R$;S$
90:END
```

LF

FORMAT: 1. LF [<lignes>]

Abréviation:
Voir également: CSIZE

BUT: Fait avancer le papier dans l'imprimante, du nombre spécifié de lignes.

REMARQUES En mode de texte, LF toute seule fait avancer le papier d'une ligne chaque fois qu'elle est exécutée. Le paramètre <lignes> spécifie le nombre de lignes compris dans la gamme spécifiée par l'instruction PAPER en cours d'exécution. Si le paramètre <lignes> est positif, le^papier se déplacera en sens direct, et s'il est négatif, le papier se déplacera en sens inverse.

La hauteur de ligne effective pour la commande LF dépend de la taille de caractère sélectionnée par CSIZE.

EXEMPLE:

```
>LF 3
```

Fait avancer le papier de trois lignes

LFILES

FORMAT: 1. LFILES "<d:>"
2.LFILES "<d:nom de fichier>"
3.LFILES "<d:nom de fichier ambigu>"

Abréviation: LF.
Voir également: FILES, SETDEV

BUT: Liste l'information de répertoire sur les fichiers spécifiés pour l'unité spécifiée sur l'imprimante ou sur l'un des ports série.

REMARQUES: Les options pour la commande LFILES sont spécifiées de la même manière que la commande FILES, mais l'information de répertoire est envoyée qu port série ou à l'imprimante au lieu d'être envoyée à l'écran.

L'accès utilisé sera celui spécifié dans la commande SETDEV en cours d'exécution, à savoir COM1: (port RS-232C) ou COM2: (port optique SIO). Si SETDEV n'est pas utilisée, l'unité par défaut est l'imprimante.

EXEMPLE: Voir FILES.

LINE

ion: LIN. *Voir également:* LLINE

FORMAT: 1. LINE[(X1,Y1)] – (X2,Y2)[,<basculement de points>]
[,<configuration<] { [,B] / [,BF] }

BUT: Trace une ligne entre deux points sur l'écran graphique.

REMARQUES La commande LINE trace une ligne à partir du point spécifié par les valeurs de coordonnées X1,Y1; ou à partir de la position actuelle du curseur graphique si les valeurs X1 et Y1 ne sont pas spécifiées, jusqu'au point spécifié par les valeurs de coordonnées X2,Y2. Les valeurs de coordonnées sont des distances à partir de la position d'origine, à l'extrémité gauche en haut sur l'écran (0,0). Le paramètre optionnel <basculement de points> change la configuration des points spécifiée dans le paramètre <configuration> de la façon suivante:

basculement de points>	Effet
S	Spécifie que les bits mis à 1 dans le paramètre <configuration> mettent les points en circuit (noirs) et que les bits mis à 0 mettent les points hors circuit.
R	Spécifie que les bits mis à 1 dans le paramètre <configuration> mettent les points hors circuit et que les bits mis à 0 mettent les points en circuit. Ceci est l'inverse de l'option S et peut être utilisé pour tracer une ligne en vision inverse sur un écran noir (tous les points en circuit).
X	Inverse l'état de tous les points le long de la ligne.

Le paramètre de basculement par défaut est S.

Le paramètre optionnel <configuration> spécifie la configuration des points utilisée pour tracer la ligne sous forme d'un nombre compris dans une gamme de 0 à 65535. La valeur décimale est alors convertie en nombre binaire à 16 bits, dont chacun représente la présence (1) ou l'absence (0) d'un point

dans cet emplacement. La valeur peut être spécifiée sous forme d'un nombre hexadécimal précédé par un signe "&", compris dans une gamme de &0000 à &FFFF.

Exemple 1 <configuration> = 65535 en valeur décimale ou &FFFF en valeur hexadécimale Valeur binaire = 1111111111111111 Configuration de ligne: xxxxxxxxxxxxxxxx —

Ceci est une ligné pleine.

Exemple 2 <configuration> = 43690 en valeur décimale ou &AAAA en valeur hexadécimale Valeur binaire = 1010101010101010 Configuration de ligne: x x x x x x x x —

Ceci est une ligne pointillée.

Exemple 3: <configuration> = 26214 en valeur décimale ou &6666 en valeur hexadécimale Valeur binaire = 0110011001100110 Configuration de ligne: xx xx xx xx —

Ceci est une ligne discontinue.

La configuration par défaut est une ligne pleine. L'option B trace une case sur la ligne sous forme d'une diagonale. L'option BF trace une case et la remplit.

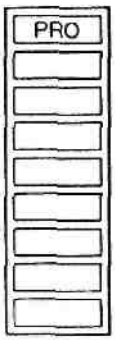
NOTE:
Les valeurs de coordonnées X et Y pour la commande LINE peuvent être spécifiées dans la gamme allant de -32768 à 32767. Mais seules les valeurs de coordonnées comprises dans une gamme de 0 à 155 pour X et dans une gamme de 0 à 31 pour Y seront affichées sur l'écran.

EXEMPLE:

```
10:CLS
20:FOR N=10 TO 100 STEP 30
30:M=N+20
40:LINE (N,10)-(M,20),,BF
50:NEXT N
60 END
```

[10] Efface l'écran.
[20] Tourne sur une boucle pour répéter le tracé de chaque case. Quatre cases apparaîtront en lignes à travers l'écran.
(30] Trace une case rectangulaire remplie.

LIST



FORMAT: 1. LIST
2. LIST<numéro de ligne>

Abréviation: L.
Voir également: LLIST

BUT; Affiche tout ou une partie d'un programme en mémoire.

REMARQUES L'exécution de LIST sans spécifier un numéro de ligne listera un écran complet du programme en cours d'exécution en partant du numéro de la ligne le plus bas. La touche H fera remonter la ligne suivante dans le listage.

LIST <numéro de ligne> affiche la ligne de programme spécifiée sur l'écran. Si le numéro de ligne spécifié est supérieur à tout numéro de ligne existant, un code d'erreur sera produit.

EXEMPLE:

```
>L I ST
```

Affiche les lignes du programme dans l'ordre

```
>L I ST  100
```

Affiche la ligne 100 du programme

LUNE

FORMAT: 1. LL!NE[(X1,Y1)]-(X2,Y2)[-(X3,Y3)..],[<type>],[<couleur>][,B]

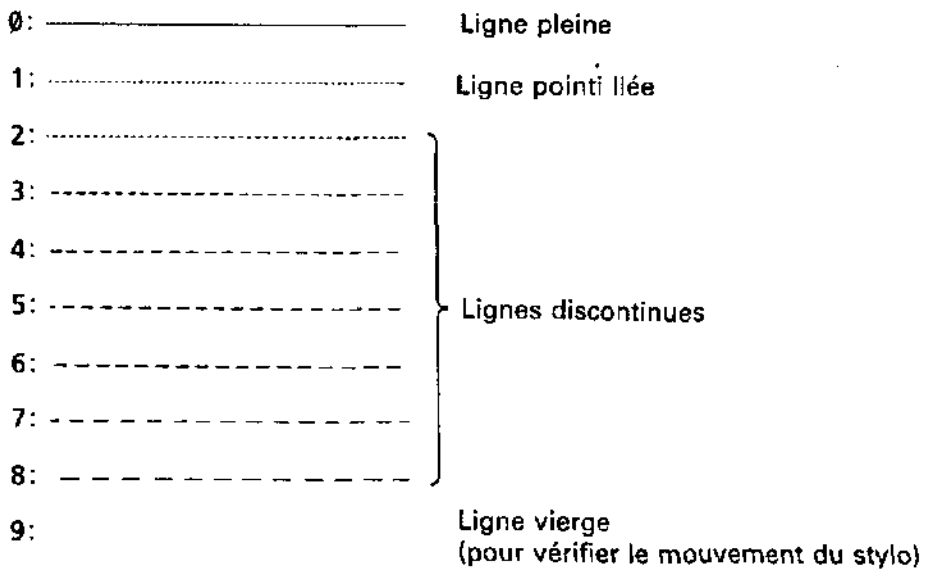
Abréviation: LLIN.
Voir également: COLOR, RLINE, SORGN

BUT: Trace une ligne ou une série de segments de ligne entre les points spécifiés par les coordonnées absolues sur l'imprimante.

REMARQUES ,

En mode graphique, la commande LLINE trace un segment de ligne à partir du point spécifié par les coordonnées X1, Y1 ou à partir de la position actuelle de la plume si les valeurs X1 et Y1 ne sont pas spécifiées, jusqu'au point spécifié par les coordonnées X2,Y2. Les valeurs X et Y doivent être comprises dans une gamme de -2048 à 2047. Les valeurs de coordonnées sont absolues, c'est-à-dire, qu'elles représentent les distances à partir du point désigné en tant qu'orgine (0,0) par l'instruction SORGN en cours d'exécution. Cinq segments de ligne contigùs subséquents peuvent être spécifiés.

Le paramètre optionnel <type> spécifie le type de ligne dans la gamme allant de 0 à 9.

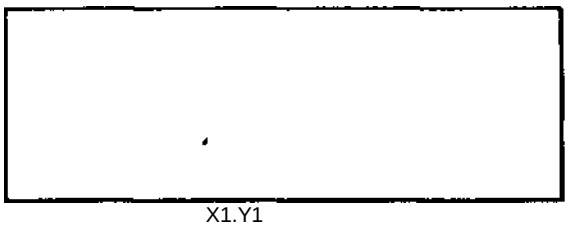


Le paramètre optionnel <couleur> spécifié la couleur de stylo dans la gamme allant de 0 à 3 (voir commande COLOR). Si les paramètres <type> et <couleur> ne sont pas spécifiés, les valeurs actuellement désignées seront utilisées.

Lors de l'exécution de LLINE ou RLINE, immédiatement après une commande LPRINT en mode graphique, le paramètre type doit être spécifié; le type de ligne par défaut ne s'applique pas dans ces cas. Ceci est également valable pour les programmes transférés depuis le PC-1500, impliquant la commande LINE du PC-1500.

L'option B spécifie un rectangle à tracer sur la diagonale spécifiée par X1,Y1 et X2,Y2 comme indiqué ci-dessous:

Pour la commande LLINE (X1,Y1) - (X2,Y2),„B



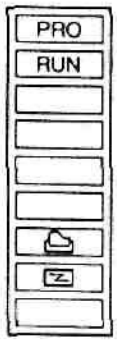
EXEMPLE:

```
10:GRAPH
20:GLCURSOR (40,40)
30:SORGN
40:LLINE -(100,0)-(0,100)-(0,0)
50:TEXT
60:END
```

Ceci tracera un triangle éloigné du coin de la page.

[10] Met la machine en mode graphique.
(20) Déplace le stylo d'imprimante jusqu'à la position spécifiée éloignée du coin de la page.
[30] Met l'origine à zéro en utilisant la position actuelle du stylo comme la position 0,0. Tous les tracés subséquents utiliseront ce système de coordonnées.
[40] Trace le triangle en trois mouvements aboutissant à l'origine.
[50] Maintenant que le tracé est terminé, remet la machine en mode de texte.

LLIST / LLIST*



FORMAT: 1. LLIST[*J
2.LLIST[*]<numéro de ligne>
3.LLIST[*]<numéro de ligne>,<numéro de ligne>
4.LLIST[*]<numéro de ligne>,
5.LLIST[*],<numéro de ligne>

Abréviation: LL. *Voir égaiement:* LIST

BUT: Liste les lignes de programme en mémoire sur l'imprimante optionnelle ou sur l'interface sérielle d'E/S si celle-ci est ouverte.

REMARQUES: La commande LLIST est utilisée de la même manière que LIST, mais la sortie est dirigée'vers l'imprimante ou l'interface sérielle d'E/S au lieu de l'écran. Elle est plus souple que la commande LIST.

Sur l'imprimante:
LLIST sans aucun paramètre liste le programme entier.
LLIST <numéro de ligne> liste seulement la ligne spécifiée du programme.
LLIST <numéro de ligne>, <numéro de ligne> liste les lignes du programme partant du premier numéro de ligne spécifié et se terminant par le second numéro de ligne spécifié.
LLIST <numéro de ligne>, liste les lignes du programme partant du numéro de ligne spécifié jusqu'à la fin du programme.
LLIST, <numéro de ligne> liste les lignes du programme jusqu'au numéro de ligne spécifié inclus.

Lors de l'utilisation de LLIST pour envoyer les lignes à l'imprimante, la sortie sur l'imprimante est affectée par le paramètre <longueur de ligne> désigné par l'instruction PCONSOLE en cours d'exécution. Si le paramètre <longueur de ligne> est fixé à 18 ou plus, les lignes seront imprimées et continuées à la ligne suivante lorsqu'elles sont trop longues. Si le paramètre <longueur de ligne> est fixé à 17 ou 16, ERROR 76 s'affichera lorsque la ligne à imprimer est plus longue qu'au paramètre <longueur de ligne> et aucune impression ne sera obtenue.

Sur l'interface sérielle d'E/S:
Si l'interface sérielle d'E/S est ouverte pour la sortie par SETDEV utilisant le paramètre PO, la commande LLIST enverra la sortie des lignes du programme à l'interface au lieu de l'envoyer à l'imprimante.

Chaque ligne dans le programme contiendra le nombre de caractères spécifié dans la commande PCONSOLE. Chaque ligne se terminera par CR, LF ou LF+CR suivant la spécification dans la commande PCONSOLE.

Si un mot de passe est désigné, LLIST est ignorée. LUST* toute seule fait sortir les lignes de commentaires commençant par une apostrophe ('). Le numéro de ligne et l'apostrophe seront supprimés. Les commentaires ne partant pas de la tête d'une ligne et tous les commentaires REM seront ignorés. Ceci est utile pour faire sortir simplement l'information descriptive d'un programme contenu dans les instructions de commentaires. Si la fin (80ème colonne) d'une ligne de commentaire se termine en point-virgule, tout commentaire apparaissant sur la ligne suivante ne sera pas imprimé sur une nouvelle ligne dans le listage mais continuera la ligne précédente.

Lors de l'utilisation de LLIST* pour le port série, les lignes de programmes disposant de numéros de lignes à 3 chiffres (c'est-à-dire 100 et plus) seront tronqués quelque part dans le listage au port à cause des limites de l'espace tampon. Par conséquent, pour obtenir des listages complets, assurez-vous de n'avoir que des numéros de lignes à deux chiffres seulement (de 0 à 99).

Si la taille de caractère actuelle sélectionnée par CSIZE est 1, le programme sera sorti en caractères de taille 1. Pour toutes les autres tailles de caractère (2 à 9), le listage sera effectué en caractères de taille 2. LLIST/LLIST* sélectionne automatiquement le mode de texte avant le listage.

EXEMPLE:

```
>LL I ST
```

Liste le programme entier.

```
>LL I ST    100,200
```

Liste les lignes du programme 100 à 200 incluse.

```
>LLIST    150. Liste à partir de la ligne 150 jusqu'à la fin du
```

programme.

```
>LL I ST   ,40
```

Liste le programme du début à la ligne 40 incluse.

FORMAT: 1. LN(X)

Voir également: EXP

LN

BUT: Renvoie le logarithme naturel (de base 'e') des expressions X.

REMARQUES: La valeur spécifiée pour X doit être supérieure à zéro. Pour trouver l'antilog d'un nombre, utiliser la fonction EXP.

EXEMPLE:

```
10:CLS
20:INPUT "TAPER UN NOMBRE ";X
30:PRINT "LE LOGARITHME NATUREL DE";X;" EST ";LN(X)
40:INPUT "ENCORE? O/N";A$
50:IF A$="O"THEN 20
60:END
```

LOAD/LOAD*

FORMAT: 1. LOAD"<d:nom de fichier>[,R] 2.
LOAD*"<d:nom de ficheier>"

Abréviation: LOA.
Voir également: CHAIN, LLIST* , MERGE, REM, RUN, SAVE

BUT: Charge un fichier dans la mémoire depuis une unité de disque souple, un module de disque RAM, une bande ou un des ports série.

REMARQUES: Le paramètre <d:nom de fichier> spécifie le nom de l'unité et le nom du fichier à charger. Les noms d'unité suivants peuvent être spécifiés.

S1:,S2:	Logements de modules 1 et 2
X:,Y:	Unité de disque souple
COMn:	Port série
CAS:	Bande cassette

Le nom de fichier spécifié doit exister sur l'unité.

Si l'option R est spécifiée après le chargement en mémoire d'un fichier à l'aide de la commande LOAD, il sera immédiatement exécuté comme si la commande RUN avait été introduite. L'option R est également utilisée avec les fichiers AUTORUN (voyez la page 80) pour permettre à un fichier d'être chargé et exécuté automatiquement. Si le fichier ne contient pas de programme BASIC, une erreur d'exécution se produira. Tous les fichiers ouverts quand LOAD est exécutée sur un autre fichier seront alors fermés sauf quand l'option R est spécifiée.

Lorsqu'un astérisque (*) est placé après LOAD dans le format 2, chaque ligne sera précédée par un numéro de ligne et une apostrophe (') après chargement en mémoire. Ceci a pour effet de faire de chaque ligne une ligne de commentaire BASIC (voir l'instruction REM). Les lignes sont numérotées à partir de 10, par échelons de 10. L'utilisation de la forme LOAD* permet aux fichiers ASCII d'être chargés dans la mémoire sous forme des lignes de commentaire BASIC. Ceci est le seul moyen pour que les données de texte en lignes puissent être traitées par le PC-1600 au niveau du fichier. La commande LLIST* peut alors

être utilisée pour lister le fichier. Par exemple, un fichier contenant les données de texte suivantes sera chargé sous la forme indiquée par l'option LOAD*.

```
10 ' L'instruction LOAD* valide les données de texte
 20 ' devant être chargées dans l'ordinateur au
30 ' niveau BASIC et affichées comme
40 ' information sur l'écran. Celle-ci peut
 50 ' être utilisée pour des notes simples, ou des
60 ' listes des adresses, etc.
```

Cette option LOAD* ne nécessite pas les fichiers de programmes BASIC, qui contiennent déjà les numéros de ligne.

EXEMPLE:

```
>LOAD"BIOCALC",R
```

Charge le programme en mémoire actuellement contenue sous le nom "BIOCALC" dans la mémoire de l'utilisateur et déclenche l'exécution du programme.

LOC

FORMAT: 1. LOC(<numéro de fichier>)

Abréviation: Voir également:

BUT: Renvoie le nombre des enregistrements qui ont été lus ou écrits depuis l'ouverture du fichier.

REMARQUES: LOC ne peut être utilisée que pour les fichiers sur le disque souple ou sur un module dans les logements 1 ou 2. Une instruction LOC peut être utilisée pour commander le déroulement du programme suivant le nombre des enregistrements qui ont été sollicités par un programme depuis l'ouverture du fichier.

EXEMPLE:

```
5:MAXFILES=1
10:OPEN "X:FICHIER01"FOR INPUT AS #1
20:IF EOF (1)THEN 50
30:INPUT #1,N
40:GOTO 20
50:M=LOC (1)
60:PRINT "LE FICHIERA";M;" ENREGISTREMENTS"
70:CLOSE #1
80:END
```

Ce programme contrôle le nombre d'enregistrements dans un fichier sur le disque souple.

[5] Affecte un espace en mémoire pour le fichier 1.
[10] Ouvre le fichier FICHIER 01 pour l'introduction.
[20] Examine si la fin du fichier est atteinte.
[30] Introduit l'enregistrement suivant. Le fichier se compose simplement des enregistrements numériques.
[40] Passe à l'enregistrement suivant et le lit.
[50-60] La fin du fichier est atteinte et LOC contient le nombre d'enregistrements lus. [70] Ferme le fichier.

LOCK / UNLOCK

FORMAT: 1. LOCK
2. UNLOCK

Abréviation: LOC./UN.
Voir également:

BUT: Invalide et revalide l'utilisation de la touche MODE.

REMARQUES: LOCK invalide la touche MODE, évitant le changement accidentel du mode. L'ordinateur est verrouillé en mode en cours. UNLOCK revalide la touche MODE pour l'utilisation normale.

EXEMPLE:

```
îO:LOCK
:
:
:
90:UNLOCK
100:END
```

[10] Invalide la touche MODE au début du programme. [90] Revalide la touche MODE à la fin.

LOF

FORMAT: 1. LOF(<numéro de fichier>)

Abréviation:
Voir également: DSKF

BUT: Renvoie le volume d'un fichier en octets.

REMARQUES LOF renvoie le volume d'un fichier sur le disque souple (X:,Y:) ou sur le module dans des logements d'extension 1 (S1:) ou 2 (S2:) en octets après qu'il a été ouvert en cours d'exécution du programme. Le paramètre <numéro de fichier> est le numéro sous lequel le fichier a été ouvert par la commande OPEN.

EXEMPLE: Ce programme contrôle le volume du fichier FILE01 sur le disque souple

```
5:MAXFILES =1
10:OPEN "X:FICHIER01"FOR INPUT AS #1
20:N=LOF (1)
30:PRINT "LE VOLUME EN OCTETS DU FICHIER01 EST DE";
40:CLOSE #1
50:END
```

15] Affecte un espace en mémoire pour le fichier 01.
[10] Ouvre le fichier FILE01 pour l'introduction.
[20-30] Trouve le volume du fichier et imprime la valeur.
[40] Ferme le fichier.

FORMAT: 1. LOG(X)

Abréviation: LO. *Voir également:*

LO
G

BUT: Renvoie le logarithme commun (de base 10) de l'expression X.

REMARQUES: Pour obtenir un logarithme d'une base autre que 10, utiliser la formule de conversion suivante: Log de base B de X = LOG(X)/ LOG(B).

Pour obtenir l'antilog d'un logarithme commun, élveer 10 à la puissance du logarithme en utilisant l'opérateur "^".

EXEMPLE:

```
>PRINT LOG(2)
             3.010299957E-01
```

LPRINT / LPRINT USING

FORMAT: 1. LPRINT[<liste des expressions>][;]
2.LPRINT USING <chaîne de format>;<liste des expressions>
3.LPRINT TAB<colonne>;<expression>;TAB<colonne>;
<expression>;TAB<.....

Abréviation: LP.
Voir également: PCONSOLE, PRINT, PZONE, TAB.

BUT: Utilisée pour sortir les données sur l'imprimante ou sur un port série.

REMARQUES: Les instructions LPRINT et LPRINT USING sont utilisées de la même manière que les instructions PRINT et PRINT USING, mais elles dirigent la sortie vers l'imprimante (ou l'un des ports série E/S) au Heu de l'écran.

LPRINT, utilisée sans aucun paramètre spécifié, fait sortir une ligne en blanc sur l'imprimante; ceci exerce le même effet qu'un changement de ligne. Cette commande peut être utilisée plusieurs fois successives pour faire avancer le papier sur l'imprimante.

LPRINT <liste des expressions> imprime les valeurs des expressions dans le paramètre <liste des expressions>. Les expressions dans la liste peuvent être des variables numériques ou alphanumériques, ou des chaînes réelles si elles sont délimitées par des guillemets. Le format exact d'impression dépend du séparateur utilisé entre les articles dans le paramètre <liste des expressions>. Si un point-virgule (;) est utilisé pour délimiter les articles dans la liste, ceux-ci sont imprimés l'un immédiatement après l'autre. Si une virgule (,) est utilisée, les articles sont imprimés à l'état positionné dans chaque zone d'impression tel que spécifié dans l'instruction PZONE. Voir la description de l'instruction PZONE pour l'explication des zones d'impression.

Si la liste des expressions est terminée par un point-virgule, la sortie suivante sur l'imprimante continuera sur la même ligne tandis que, si la liste est terminée sans un point-virgule, la sortie suivante sera imprimée sur une nouvelle ligne.

L'option USING <chame de format> est spécifiée de la même façon que pour l'instruction PRINT USING. Se référer à la description pour l'explication détaillée de la méthode de spécification des chaînes de format.

L'option TAB spécifie la colonne réelle dans laquelle l'article suivant doit être imprimé. Si la colonne TAB est supérieure à la largeur d'impression spécifiée dans l'instruction PCQNSOLE en cours d'exécution, un code d'erreur sera produit.

Si l'unité spécifiée dans l'instruction SETDEV en cours d'exécution est l'un des ports série, l'instruction fera sortir sur le port au lieu de l'imprimante.

Lorsque LPRINT est spécifiée en mode graphique, les lignes ne sont pas terminées par CR et LF.

Voir PRINT USING et PZONE.

MAXFILES

FORMAT: 1. MAXFILES = <nombre de fichiers>

Abréviation: MA.
Voir également: CLOSE, OPEN

BUT: Spécifie le nombre maximum de fichiers pouvant être ouverts à la fois.

REMARQUE : Tous les fichiers doivent être fermés au moment où la commande MAXFILES est exécutée. Le paramètre <nombre de fichiers> est 0 juste après la mise sous tension et MAXFILES doit donc être spécifiée avant qu'un fichier quelconque ne puisse être ouvert. Le <nombre de,fichiers> maximum est 15. S'il existe une mémoire insuffisante pour affecter le nombre spécifié de fichiers, une erreur sera produite. Le volume de chaque fichier affecté par MAXFILES est de 313 octets. Ce volume sera augmenté selon la nécessité dans un programme.

EXEMPLE:

```
>MAXFILES-=3
```

Permet l'ouverture simultanée de trois fichiers au maximum.

MEM

FORMAT: 1. MEM

Abréviation: M.
Voir également: STATUS

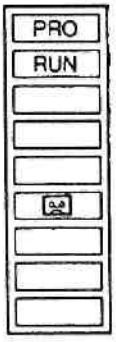
BUT: Renvoie le volume de l'espace mémoire non utilisé dans la zone utilisateur.

REMARQUES: Le volume de l'espace mémoire non utilisé est indiqué en octets, y compris la zone de mémorisation des variables (même que STATUS 0).

EXEMPLE:

```
>MEM
```

MERGE

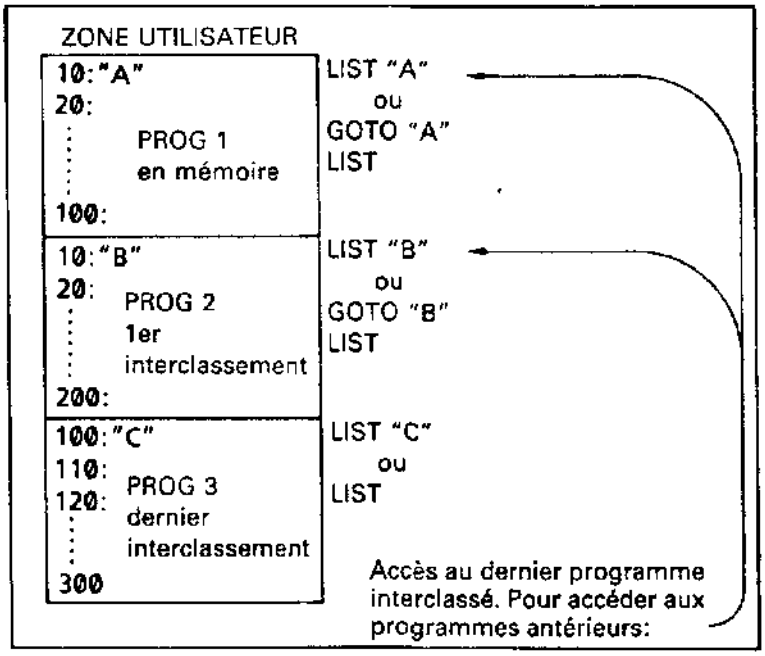


FORMAT: 1. MERGE
2. MERGE"<nom de fichier>"

Abréviation: MER. *Voir également:* CLOAD

BUT: Utilisée pour charger un programme conservé sur la bande cassette avec un autre programme actuellement en mémoire.

REMARQUES: Les programmes interclassés en mémoire peuvent avoir les mêmes numéros de ligne que les programmes existants. Chaque programme est affecté à une zone exclusive en mémoire. Seule GOTO<étiquette> ou LIST<étiquette> permettra le déplacement entre programmes. LIST listera seulement le dernier programme MERGE (interclassé). Pour lister les autres programmes, utiliser l'option <étiquette> avec LIST. Ceci signifie que tous les programmes interclassés en mémoire devront avoir les étiquettes sur leurs premières lignes.



De plus, faites attention lorsque les programmes d'interclassement contenant des instructions READ et DATA. Afin que l'instruction READ accède aux instructions DATA correctes après l'interclassement, faites précéder la ligne DATA d'un paramètre <étiquette>. Ensuite, insérer une ligne avant l'instruction READ en y spécifiant <étiquette> RESTORE comme le montre l'exemple ci-dessous:

```
10  RESTORE "A"
20  READX,Y,Z
30  PRINTX;Y;Z
40  "A":DATA 3,4,5
```

MERGE seule interclasse le programme suivant sur bande depuis la position actuelle de la bande dans la mémoire. Le paramètre <nom de fichier> spécifie lequel des fichiers contenus sur bande doit être MERGE 1 (interclassé).

EXEMPLE:

```
>MERGE "PROFIT"
```

Recherche le programme "PROFIT" sur la bande en partant de la position actuelle de la bande et le charge dans la mémoire d'utilisateur en cours.

MID$

FORMAT: 1. MID$(X$,N,M)
2. MID$("<chaîne>",N,M)

Abréviation: Ml.
Voir également: LEFT$, RIGHT$

BUT: Renvoie une chaîne de M caractères depuis la chaîne intérieure X$ partant du N^{ème} caractère dans la chaîne X$.

REMARQUES : Si N est inférieur à 1 ou supérieur au nombre de caractères dans X$, une chaîne vide sera renvoyée. M doit être compris dans une gamme de 0 à 80 et N, dans une gamme de 1 à 80. Les fractions seront arrondies à l'unité inférieure (tronquées).

EXEMPLE:

```
10:Z$="ABCDEFG"
20:LET Y$=MID$ (Z$,3,4)
30:PRINT Y$

>RUN
CDEF
>
```


FORMAT: 1. <n>MOD<m>

Abréviation:
Voir également: INT

MO
D

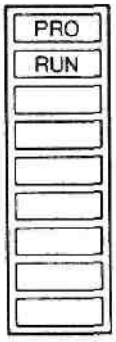
BUT: Renvoie le reste de la division de <n> par <m> où n et m sont des nombres entiers décimaux. Si n ou m ne sont pas des entiers, ils seront arrondis au nombre entier le plus proche avant la division.

EXEMPLE:

```
10:INPUT "LA VALEUR DE n ";N
20:INPUT "LA VALEUR DE m ";M
30:A=N MOD M
40:PRINT:"LE RESTE EST ";A
50:GOTO 10
60:END
```

[10-20] Introduit les paramètres.
[30] Calcule le reste. [40] Imprime
la réponse.
[50] Répète le processus. L'utilisateur doit appuyer sur la touche BREAK pour arrêter le programme.

MODE



FORMAT: 1. MODE [0] 2.
MODE 1

Abréviation: MO.
Voir également:

BUT: Sélectionne le mode d'écran.

REMARQUES: La commande MODE ne peut pas être spécifiée sur une ligne de programme; elle doit être introduite en mode direct.

Si MODE ou MODE 0 est spécifiée, toutes les quatre lignes sur l'écran sont en activité, et l'ordinateur peut être utilisé avec CE-1600P et d'autres périphériques pour PC-1600. Le caractère désigné pour PC-1600 est sélectionné.

MODE 1 sélectionne le mode PC-1500. En ce mode, seule la ligne inférieure de l'écran est en activité, mais l'écran fait défiler les lignes une à une vers le haut lorsque l'ordinateur se trouve en mode PRO pour l'introduction ou l'édition d'un programme ou lors de l'utilisation de la commande INPUT. La longueur de la ligne sur l'écran est fixée à 26 caractères, et l'ordinateur peut être utilisé avec les périphériques PC-1500. Le caractère désigné pour PC-1500 est également sélectionné. Voir l'Annexe C pour les tables de code de caractères dans les deux modes susmentionnés. Il n'est pas possible d'exécuter les programmes qui ont été créés sur PC-1500 en MODE 0; ils doivent être exécutés en MODE 1.

En MODE 0, les instructions PRINT affichent les données sur les lignes consécutives de l'écran jusqu'à remplir l'écran. Ensuite, l'écran réalise un défilement d'une ligne à la fois. Les articles plus longs qu'une ligne de l'écran (26 caractères) sont portés à la ligne suivante.

En MODE 1, les instructions PRINT affichent les données sur la ligne inférieure de l'écran. Les lignes comportant plus de 26 caractères seront tronquées, et la partie tronquée est perdue. Les instructions PRINT subséquentes recouvrent chaque fois l'affichage existant. L'écran ne peut pas réaliser le défilement sauf lors de l'introduction ou de l'édition d'un programme ou lors de l'utilisation d'une instruction INPUT.

NOTE:
Après le passage d'un mode à l'autre, le caractère de sollicitation sera affiché à la gauche de l'écran en mode RUN, et le curseur sera affiché en position d'origine (0,0) sans effacer l'écran en modes PRO et RESERVE. En mode PRO, toutes les quatre lignes de l'écran sont en activité tant en MODE 0 qu'en MODE 1.

EXEMPLE:

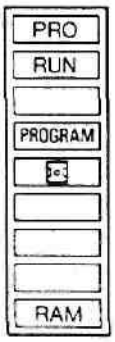
```
>M0DE
```

Sélectionne le mode PC-1600.

```
>MODE  1
```

Sélectionne le mode PC-1500.

FORMAT: 1. NAME "<d:ancien nom de fichier>"
AS"<d:nouveau nom de fichier>"



Abréviation: NA.
Voir également: COPY, FILES

BUT: Change le nom d'un fichier sur le disque souple ou le disque RAM.

REMARQUES: Le paramètre <d:ancien nom de fichier> doit être le nom d'un fichier qui existe sur l'unité spécifiée, par d:. Le paramètre <d:nouveau nom de fichier> doit être un nom de fichier qui n'est assigné à aucun autre fichier sur l'unité d:.

Le même nom d'unité doit être spécifié pour le fichier doté d'un nouveau nom.

Le changement de nom de fichier n'est pas effectif dans les cas suivants:

(1)Le disque souple dans l'unité spécifiée dans d: a son encoche de protection en écriture couverte.
(2)Le fichier spécifié avec le paramètre <d:ancien nom de fichier> a été protégé en écriture par l'option P de la commande SET.
(3) Le fichier spécifié est actuellement ouvert.
(4) L'interrupteur de protection en l'écriture est mis en position ON sur le module de disque RAM.

EXEMPLE:

```
>NAME "XIAMCIEM MOM" "NOUVEAU NOM"
```

Change le nom de fichier OLDNAM sur le disque souple en NEWNAM.

FORMAT: 1. NEW
2.NEW "Sn:"[,<adresse>]
3.NEW <adresse>
4.NEW 0

Abréviation:

Voir également: DELETE, STATUS, TITLE

BUT: Efface toutes les lignes de programme actuellement en mémoire, et/ou affecte l'espace mémoire pour les programmes de code machine.

REMARQUES Si elle est utilisée en mode RESERVE, elle efface toutes les affectations des touches de fonction de la mémoire de réserve. Si un programme est introduit en mémoire par clavier avant d'effacer les lignes de programme existantes, celles portant les numéros différents de celles introduites par clavier seront maintenues telles quelles, causant des problèmes graves au niveau de la programmation. L'utilisation de NEW évite cet inconvénient en premier lieu.

Mode PC-1600 (MODE 0)
La zone accessible en mémoire de l'ordinateur est désignée comme la "zone de l'utilisateur". La zone de l'utilisateur dans le PC-1600 est d'un volume de 11834 octets, et inclut la zone de programme BASIC, la zone de stockage des variables et la zone libre. Toute mémoire affectée aux programmes de langage machine par la commande NEW réduit le volume utile de la zone de l'utilisateur. Les adresses utilisées sont des adresses relatives par rapport au haut de la mémoire (0).

NEW efface tous les programmes BASIC et machine dans la mémoire en cours qui a été spécifiée en utilisant TITLE et laisse la zone affectée aux programmes de langage machine à la valeur préalablement désignée.

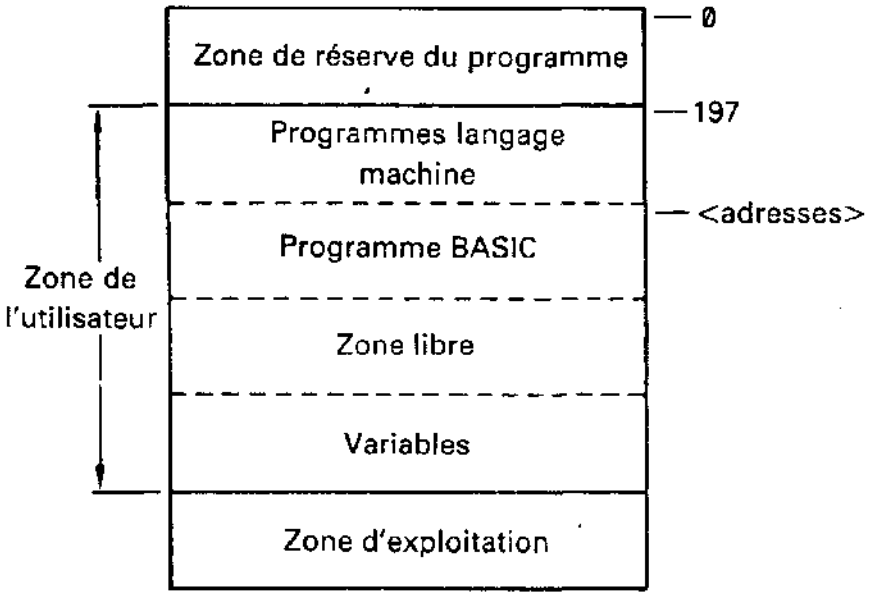
NEW "Sn:" efface tous les programmes dans la zone en mémoire désignée par Sn, où:

S est la zone principale en mémoire de l'ordinateur.
0 est le module de programme actuellement dans le logement 1
S est le module de programme actuellement dans le logement 2
1
S
2

et laisse la zone affectée aux programmes de langage machine à la valeur préalablement désignée.

NEW "Sn:",<adresse> efface tous les programmes dans la zone en mémoire désignée par Sn et affecte l'adresse supérieure de la zone de programme de langage machine au paramètre <adresse>. L'adresse inférieure est 197. L'adresse supérieure est 65535.

Affectation de la RAM interne:



NEW 0 efface tous les programmes BASIC et de langage machine dans la zone en mémoire spécifiée par la commande TITLE et désigne la zone affectée au programme de langage machine à 0 (adresse = 197).

Mode PC-1500 (MODE = 1)
NEW efface tous les programmes dans la zone utilisateur et laisse la zone affectée aux programmes de langage machine à la valeur préalablement désignée.

NEW <adresse> efface tous les programmes dans la zone utilisateur et affecte l'adresse supérieure de la zone de programme de langage machine au paramètre <adresse>. L'adresse inférieure est de 197. NEW 0 efface tous les programmes dans la zone utilisateur et désigne la zone affectée aux programmes de langage machine à 0 (adresse = 197).

EXEMPLF-

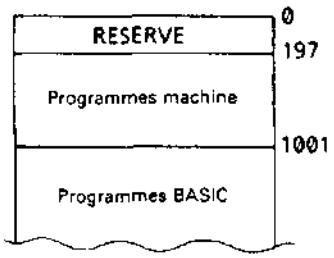
```
>LIST
10:REM VOICI UN
20:REM PROGRAMME DE TROIS LIGNES
30:END
>

>NEW
>LIST
>


>NEW 1001
```

NEW efface la mémoire, et le programme est perdu. '

Ceci efface également la mémoire en cours mais réserve 'es adresses 197 a 1000 pour les programmes de lanaaqe machine.



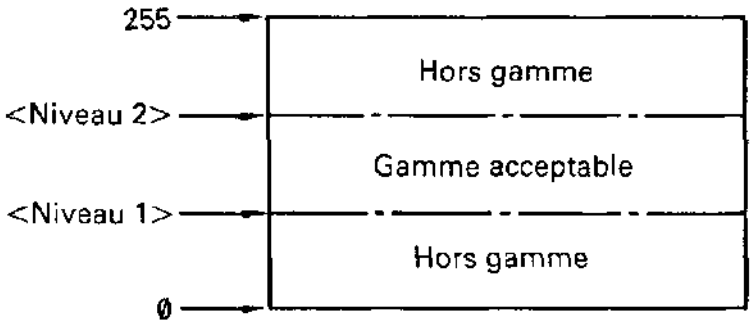
ON ADIN GOSUB

FORMAT: 1. ON ADIN(<niveau 1>,<niveau 2>)
GOSUB<numéro de ligne/étiquette>

Abréviation: 0. AD. GOS.
Voir également: ADIN ON/OFF/STOP, AIN, RETI

BUT: Provoque le branchement de l'exécution sur le numéro de ligne spécifié lorsque la tension d'entrée à la borne d'entrée analogique dévie d'une gamme désignée.

REMARQUES: Les paramètres <niveau 1> et <niveau 2> spécifient la gamme entre 0 et 255 à l'extérieur de laquelle l'entrée produira une interruption de la façon suivante:



0 ⩽ niveau 1 < niveau 2 ⩽ 255

Se référer à la Section 6 de la 3ème partie en ce qui concerne les détails des niveaux d'entrée.

L'introduction d'une interruption dans l'instruction ON ADIN GOSUB peut être commandée par l'instruction ADIN ON/OFF/ STOP. Si l'instruction ADIN ON n'est pas spécifiée dans le programme après l'instruction ON ADIN GOSUB, la désignation par défaut est ADIN STOP, et celle-ci sera effective après l'exécution de l'instruction invalidant toutes les interruptions.

Le sous-programme indiqué dans cette instruction doit être terminé par une instruction RETI.

Le maximum d'interruptions autorisées dans un programme est de 8.

[20] Définit l'endroit où s'effectue un branchement lors de la réception d'un signal à la borne d'entrée non compris dans la gamme spécifiée.

EXEMPLE:

```
20 ON ADIN (103,206)GOSUB 500
:
:
:
:
420 END
500 REM SOUS-PROGRAMME D'INTERRUPTION DE L'ENTREE DE BORNE
:
:
:
600 RETI
```

[30-480] Programme principal. Lorsque l'interruption a lieu en cours d'exécution de ces lignes, un branchement s'effectue immédiatement sur le début du sous-programme sur la ligne 500. [500-600] Sous-programme d'interruption de l'entrée de borne. L'action est prise suivant l'interruption. Le sous-programme se termine en instruction RETI. Son effet est équivalent à RETURN dans un
sous-programme normal.

ON COMn GOSUB

FORMAT: 1. ON COMn GOSUB<numéro de ligne/étiquette>

Abréviation: 0. COM GOS.
Voir également: COMn ON/OFF/STOP, RETI

BUT: Cause un branchement de l'exécution sur la ligne spécifiée d'un sous-programme lorsqu'une interruption est reçue au port spécifié.

REMARQUES Le paramètre n spécifie le port RS-232C (n = 1) ou port série optique E/S (n=2). L'introduction d'une interruption dans l'instruction ON COMn GOSUB peut être commandée par l'instruction COMn ON/OFF/STÇP. Si l'instruction COMn ON n'est pas spécifiée dans le programme après l'instruction ON COMn GOSUB, la désignation par défaut est COMn STOP, et celle-ci sera valide après l'exécution de l'instruction, invalidant toutes les interruptions.

L'indication du sous-programme dans cette instruction doit se terminer dans une instruction RETI. Le maximum d'interruptions autorisées dans un programme est de 8.

NOTE:

Voir la Partie III, Chapitre 6 pour les détails d'utilisation des signaux de commande pour les interruptions aux deux ports série.

EXEMPLE:

```
20:ON COM1 GOSUB 500
:
:
:
490:END
500:REM INTERRUPT ROUTINE FOR RS-232C PORT
:
:
:
600:RETI
```

(20] Définit où s'effectue un branchement lors de la réception d'une interruption COM1. [20-480] Programme principal. Si une interruption C0M1 quelconque est reçue en cours d'exécution de ces lignes, un branchement s'effectue immédiatement sur le début du sous-programme sur la ligne 500. [500-600] Sous-programme d'interruption du port série COM1 ou RS-232C. L'action est prise suivant l'interruption de COM1. Le sous-programme se termine dans l'instruction RETI. Son effet est équivalent à RETURN dans un sous-programme normal.

ON ERROR GOTO

FORMAT: 1. ON ERROR GOTO <numéro de ligne/étiquette>

Abréviation: 0. ER. G.
Voir également: RESUME, ERL, ERN

BUT: Provoque le branchement de l'exécution sur la ligne spécifiée (ordinairement sur la première ligne d'un sous-programme de traitement d'erreurs) lorsqu'une erreur se produit.

REMARQUES Les sous-programmes de traitement d'erreurs peuvent être écrits par l'utilisateur pour évaluer quel type d'erreur s'est produit (par exemple, une erreur de syntaxe ou une erreur de format de données) ou pour recherqher l'apparition éventuelle d'un certain type d'erreur et pour ensuite reprendre l'exécution à un point différent dans le programme suivant le type d'erreur. S'il existe une erreur dans le programme de traitement d'erreurs lui-même, BASIC revient à l'instruction ON ERROR GOTO, affiche le code d'erreur et termine l'exécution.

Le sous-programme de traitement d'erreurs doit se terminer sur RESUME, STOP ou END. Il n'existe aucune limite pour le nombre d'instructions ON ERROR GOTO dans un programme. Lorsqu'une erreur est produite, l'exécution passe suivant la spécification de la dernière instruction ON ERROR GOTO qui a été exécutée. ON ERROR GOTO 0 revient au traitement d'erreurs normal en BASIC. L'effet de ON ERROR GOTO est annulé par l'exécution de RUN et END ou par l'opération de suppression totale par les touches Miliai + El mais non au début de l'exécution par GOTO ou DEF.

EXEMPLE:

(100-140) Sous-programme de traitement d'erreurs pour rechercher l'Erreur 160.

```
5:ON ERROR GOTO 100
10:SAVE "X:DEMO"
20:PRINT "OK"
30:END
:
:
100:IF ERN=160 THEN PRINT "PAS DE DISQUE DANS L'UNITEX:"
110:PRINT "INTRODUIRE LE DISQUE ET APPUYER SUR O"
120:INPUT A$
130:IF A$="O"THEN RESUME
140:STOP
```

ON..GOSUB/ON..GOTO

FORMAT: 1. ON <expression numerique> GOTO <liste des numéros de ligne/ étiquettes> 2. ON <expression numérique> GOSUB <liste des numéros de ligne/ étiquettes>

Abréviation: 0. GOS./O.G.
Voir également: GOSUB...RETURN, GOTO

BUT: Cause un branchement de l'exécution inconditionnellement sur l'une des plusieurs lignes de programme suivant la valeur simple d'une expression lors de l'exécution.

REMARQUES :La valeur d'<expression numérique> au moment de l'exécution détermine à quel numéro de ligne hors de la <liste des numéros de ligne> l'exécution fera un branchement. Si la valeur d'<expression numérique> est 1, l'exécution fera un branchement au premier numéro de ligne dans la liste; si elle est 2, l'exécution fera un branchement au second numéro de ligne dans la liste, et ainsi de suite. Si la valeur d'<expression numérique> est supérieure au nombre d'articles dans la <liste des numéros de ligne> ou inférieur à 1, l'exécution passe à la ligne de programme suivante.

Avec la forme ON..GOSUB, chacun des.numéros de ligne dans la liste doit être la première ligne d'un sous-programme.

EXEMPLE:

```
10:INPUT "NOMBRE (1-3)?",N
20:ON N GOTO 30,50,70
30:PRINT "IGNITION"
40:GOTO 80
50:PRINT "BOOST"
60:GOTO 80
70:PRINT "ORBIT"
80:END
```

[101 Introduit un nombre de 1 à 3. Noter qu'ici, il n'existe aucun contrôle de ce que la valeur introduite par clavier est comprise dans cette gamme. Il serait prudent d'inclure une instruction IF immédiatement
après cette ligne pour assurer ce contrôle.
[20] L'exécution fait maintenant un branchement sur l'une des trois lignes suivant la valeur de N.
[30] Fait un branchement ici lorsque N est 1.
[50] Fait un branchement ici lorsque N est 2. [70] Fait
un branchement ici lorsque N est 3.

```
10:INPUT "NOMBRE (1-3)?",N
20:ON N GOSUB 100,200,300
:
90:END
100:REM PREMIER SOUS-PROGRAMME
:
190:RETURN
200:REM DEUXIEME SOUS-PROGRAMME
:
290:RETURN
300:REM TROISIEME SOUS-PROGRAMME
:
390:RETURN
```

Le principe d'utilisation de ON GOSUB est le même que ON GOTO à l'exception du fait que l'exécution passe au début d'un sous-programme.

ON KEY GOSUB

FORMAT: 1. ON KEY GOSUB <liste des numéros de ligne/étiquettes>

Abréviation: 0. KEY G OS.
Voir également: KEY ON/OFF/STOP, RETI

BUT: Cause un branchement de l'exécution du programme lorsque l'une des touches de fonction F1 à F6 est enfoncée en cours d'exécution.

REMARQUES: La touche de fonction enfoncée détermine sur quel numéro de ligne de ceux de la <liste des numéros de ligne/étiquettes> l'exécution fera un branchement. Si la touche F1 est enfoncée, l'exécution fera un branchement sur le sous-programme du premier numéro de ligne listé dans les <numéros de ligne/étiquettes>. Si la F6 est enfoncée, l'exécution se branchera sur le sous-programme du sixième numéro de ligne cité dans la liste. Si la <liste des numéros de ligne/étiquettes> contient plus de six articles, les sous-programmes des numéros de ligne dans les articles en excès ne seront jamais sollicités par cette instruction. Si la <liste des numéros de ligne/étiquettes> contient moins de six articles, l'enfoncement d'une touche de fonction qui n'a aucun numéro de ligne correspondant n'exercera aucun effet sur l'ordre d'exécution des programmes.

Chaque sous-programme doit se terminer par une instruction RETI pour renvoyer l'exécution à l'instruction immédiatement après l'instruction ON KEY GOSUB. L'utilisation des touches de fonction en tant qu'interruptions du moment de l'exécution n'est effective que pendant l'exécution du programme. L'exécution de la commande RUN ou de l'instruction END supprime toutes les affectations des touches de fonction par ON KEY GOSUB et renvoie les fonctions des touches aux désignations originales. Si une instruction KEY ON n'est pas spécifiée dans le programme après une instruction ON KEY GOSUB, la désignation par défaut est KEY STOP, et celle-ci entrera en vigueur après l'exécution de l'instruction, invalidant toutes les interruptions.

EXEMPLE:

```
:

:

:

:

20: ON KRY G0SUB 400, S00, 600 30: KEY(
1) ON•tn: KEY(2) ON50IKEY(3) ON

390: END
400:REM SOUS-PROGRAMME D'INTERRUPTION PAR LA TOUCHE 1

•t90: RETI
500:REM SOUS-PROGRAMME D'INTERRUPTION PAR LA TOUCHE 2

590: RF. TI
600:REM SOUS-PROGRAMME D'INTERRUPTION PAR LA TOUCHE 3

690: RETI
```

[20-50] Définit où la commande se branche lorsque qu'une touche est enfoncée.
[60-390] Le programme principal. Si toute interruption de touche est reçue pendant l'exécution de ces lignes, la commande fera immédiatement un branchement au commencement du sous-programme correspondant.
[400-690] Sous-programme d'interruption par touche. Les mesures sont adoptées suivant l'interruption de touche. Le sous-programme se termine par l'instruction RETI. Son effet équivant à RETURN (retour) dans un sous-programme normal.

ON PHONE GOSUB

FORMAT: 1. ON PHONE GOSUB <numéro de ligne/étiquette>

Abréviation: O. PH. GOS.
Voir également: PHONE ON/OFF/STOP, RETI

BUT: Cause un branchement de l'exécution sur le numéro de ligne spécifié lors d'une introduction à l'ordinateur depuis un modem téléphonique via le port RS-232C.

REMARQUES L'introduction d'une interruption dans l'instruction ON PHONE GOSUB peut être commandée par l'instruction PHONE ON/OFF/STOP. Si une instruction PHONE ON n'est pas spécifiée dans le programme après une instruction ON PHONE GOSUB, la désignation par défaut est PHONE STOP, et celle-ci sera valide après l'exécution de l'instruction, invalidant toutes les interruptions.

Le sous-programme cite dans cette instruction doit se terminer dans une instruction RETI. Le nombre maximum d'interruptions dans un programme est de 8.

NOTE:
Le signal réel appelé par l'ordinateur en utilisant cette instruction est le signal CI à la broche 9 de l'interface RS-232C. Ce signal est soutenu par des modems de quelques types. Voyez le Chapitre 6 de la 3 ème partie en ce qui concerne les détails des signaux de commande pour le port série RS-232C.

EXEMPLE:

```
20:ON PHONE GOSUB 500
:
:
:
490:END
500:REM PHONE INTERROMPT LE SOUS-PROGRAMME
510:VOUS AVEZ UN APPEL
:
:
:

600iRET I
```

[20] Définit où l'exécution fait un branchement lors de la réception d'une interruption téléphonique. [30-480] Programme principal. Si une interruption téléphonique quelconque est reçue en cours d'exécution de ces lignes, l'exécution fait un branchement immédiatement sur le début du sous-programme sur la ligne 500.
[500-600! Sous-programme d'interruption téléphonique. L'action est prise suivant l'interruption téléphonique. Le sous-programme se termine dans l'instruction RETI. Son effet est équivalent à RETURN dans un sous-programme normal.

ON T1ME$ GOSUB

FORMAT: 1. ON TIME$="MM/DD/HH/mm" GOSUB <numéro de ligne/étiquette>

Abréviation: 0. Tl. GOS.
Voir également: RETI, TIMES, TIME$ ON/OFF/STOP

BUT: Cause un branchement de l'exécution sur un sous-programme lorsque l'heure spécifiée est atteinte.

REMARQUES: L'horloge temps réel incorporée fournit l'entrée avec laquelle l'heure désignée dans le paramètre TIME$="MM/DD/HH/mm" est comparée. Lorsque l'heure désignée est atteinte, l'exécution fait un branchement sur la première ligne d'un sous-programme. Le sous-programme dojt être terminé sur la dernière ligne contenant l'instruction RETI.

En ce qui concerne le format sous lequel l'heure est spécifiée, se référer à la description de l'instruction TIMES.

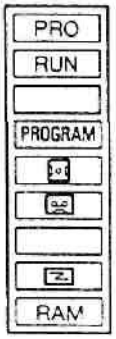
L'instruction ON TIMES GOSUB peut être invalidée et revalidée par l'instruction TIMES ON/OFF/STOP. Si l'instruction TIMES ON n'est pas spécifiée dans le programme après l'instruction ON TIMES GOSUB, la désignation par défaut est TIMES STOP, et celle-ci sera valide après l'exécution de l'instruction, invalidant toutes les interruptions. Le nombre maximum d'interruptions dans un programme est de 8.

EXEMPLE:

```
10:ON TIME$="05/12/15/30" GOSUB 500
```

(10] Fait un branchement sur le sous-programme d'interruption de la ligne 500 aux date et heure spécifiées en cours d'exécution.

OPEN



FORMAT: 1. OPEN "<d:nom de fichier>" FOR {INPUT / OUTPUT / APPEND} AS #<numéro de fichier>

Abréviation: OP.
Voir également: CLOSE, INPUT#, MAXFILES, PRINT#

BUT: Valide l'entrée et la sortie des données d'un fichier sur le disque souple ou une autre unité.

REMARQUES: Cette commande affecte à un fichier de données un numéro par lequel le fichier est spéoifié dans les instructions subséquentes de lecture ou d'écriture (voir descriptions de INPUT# et PRINT#). Après l'exécution de cette commande, le fichier est dit "ouvert".

Un fichier peut être ouvert en trois modes; INPUT, OUTPUT et APPEND. En mode INPUT, les données sont lues à partir du fichier par l'instruction INPUT#. En mode OUTPUT, les données sont écrites dans le fichier par l'instruction PRINT#. En mode APPEND, les données sont ajoutées à la fin des données du fichier existant par l'instruction PRINT#.

Le mode OUTPUT ne peut pas être utilisé pour ajouter les données à un fichier existant; il permet seulement d'écrire les données dans un nouveau fichier. Si un fichier existe déjà sous le même nom sur l'unité, il sera complètement recouvert.

Le paramètre <d:nom de fichier> spécifie le nom d'unité suivi par le nom de fichier. La valeur de <numéro de fichier> est un nombre quelconque entre un et le nombre maximum de fichiers désigné par la commande MAXFILES en cours d'exécution.

Un fichier ne peut pas être ouvert à la fois pour l'entrée et la sortie; il doit premièrement être fermé, puis ouvert à nouveau. Un fichier ne peut pas être ouvert dans les cas suivants:

(1)Le mode OUTPUT est spécifié pour un fichier qui a été protégé en écriture par l'option P de la commande SET.
(2)Le fichier devant être ouvert se trouve sur un disque souple dont l'encoche d'interdiction de l'écriture est couverte.

(3)Le fichier devant être ouvert se trouve sur un module de disque RAM dont l'interrupteur d'interdiction de l'écriture est mis en position "ON".

(4)COM1: ou COM2: (ports série E/S) sont spécifiées pour l'unité en mode APPEND.

EXEMPLE:

```
5:MAXFILES =1
10IOPEN "XIDOMNEE'FOR OUTPUT AS #1
```

```
20:FOR J=1 TO 5
30:?RINT *1,J
4O:NEXT J
50:CLOSE *1
60:0?EN "XIDONMEE" FOR INPUT AS *1
70:1F EOF (l)THEM 110
80IINPUT *1,J
90:?RINT J
10OIGOTO 70
110: REM LA FIN DU FICHIER EST ATTEINTE
12OiCLOSE #1
130:END
```

[5] Espace affecté a un fichier en mémoire
[10] Crée un nouveau fichier sur le disque.
[30] Ecrit les numéros 1 à 5 consécutivement sur le fichier.
[60] Ouvre le fichier à nouveau.
[70] Recherche la fin du fichier.
[80] Lit et imprime les numéros 1 à 5.

OUT

FORMAT: 1. OUT <adresse d'accès>,<liste des expressions>

Abréviation:

Voir également: INP

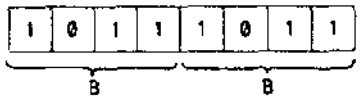
BUT: Fait sortir une configuration binaire directement sur le port de sortie du microprocesseur Z-80A.

REMARQUES: OUT fait sortir la configuration binaire pour les valeurs des expressions dans la <liste des expressions> directement sur le port de sortie machine spécifié. Le paramètre odresse du port> spécifie le port de sortie machine spécifié sous forme d'une adresse à 16 bits comprise dans une gamme de 0 à 65535 (&0 à &FFFF). Le paramètre <liste des expressions> spécifie une liste d'une ou plusieurs expressions, séparées par des virgules, dont les valeurs comprises dans une gamme de 0 à 255 (&0 à &FF) sont envoyées aux ports concécutifs en partant de l'adresse du port spécifié.

EXEMPLE:

```
> C 'J T   8 0,137
```

Fait sortir le code binaire pour 187 (&BB) à l'adresse d'accès 80 (&50). La configuration binaire &BB apparaît de la manière suivante:



OUTSTAT

FORMAT: 1. OUTSTAT "COM1:"[,<désignations>]

Abréviation: OU.

Voir également: INSTAT

BUT: Désigne les états des signaux de commande pour le port série RS-232C.

REMARQUES: OUTSTAT désigne les états des signaux RTS (demande d'envoi) et DTR (terminal de données prêt) pour le port RS-232C, de la façon suivante:

<désignation>	RTS	DTR
0	Haut	Haut
1	Haut	Bas
2	Bas	Haut
3	Bas	Bas

Si le paramètre <désignation> n'est pas spécifié, les états des signaux RTS et DTR restent hauts tandis qu'une commande de port série est en cours d'exécution ou que le port est en cours de réception de données, mais se trouvent bas à d'autres moments. Lorsque le débit de données en entrée rend le tampon de réception saturé, RTS passe à l'état bas afin d'arrêter temporairement la transmission de données et de permettre de vider le tampon.

EXEMPLE:

```
>OUTSTAT "COM1:",2
```

Désigne RTS bas et DRT haut pour le port RS-232C.

PAPER

FORMAT: 1. PAPER <type>[,<limite en sens inverse>][,<limite en sens direct>]

Abréviation: PAP.
Voir également: GRAPH, TEXT

BUT: Désigne le type de papier et la gamme d'impression dans la direction verticale pour l'imprimante.

REMARQUES: Le type de papier est spécifié de la façon suivante:

C pour les feuilles coupées R
pour le papier en rouleau

Les deux paramètres de limitation d'impression en option <limite en sens inverse> et <limite en sens direct> désignent la zone imprimable dans la direction Y (verticale) à des échelons de 0,2 mm. L'enfoncement de la touche j sur l'imprimante, la mise sous tension ou l'exécution de TEXT, GRAPH ou PAPER désignent les limites d'impression dans l'instruction PAPER à partir des positions actuelles du paper et du stylo. Lors de la mise sous tension, les dernières valeurs désignées sont effectives. Après une ALL RESET, le paramètre <type> revient à R.

Le paramètre <limite en sens inverse> spécifie la limite de la gamme d'impression en amont comprise dans une gamme de 30 à 2047. La valeur par défaut est de 30 (6 mm) pour la feuille coupée et de 999 (199,8 mm) pour le papier en rouleau. Le paramètre <limite en sens direct> spécifie la limite de la gamme d'impression en aval comprise dans une gamme de 30 à 2047. La valeur par défaut est de 1354 (272,8 mm) pour la feuille coupée et de 999 (199,8 mm) pour le papier en rouleau.

Les paramètres <limite en sens inverse> et <limite en sens direct> sont remis à leurs valeurs par défaut lorsque la commande GRAPH ou TEXT est exécutée. De plus, en mode TEXT, la valeur par défaut pour le paramètre <limite en sens direct> devient infinie lorsque le papier en rouleau est spécifié.

NOTE:

En mode graphique, une instruction pour déplacer le stylo à l'extérieur de la gamme spécifiée dans l'instruction PAPER trace la réflexion du tracé projeté lors de la désignation des limites d'impression.

EXEMPLE:

```
>PAPER C
```

Désigne l'imprimante pour l'utilisation des feuilles standard de format A4. Les valeurs par défaut de <limite en sens inverse> et de <limite en sens direct> seront utilisées.

FORMAT: 1. PASS "<chaîne>"

Abréviation: PAS.

Voir également: CLOAD, CSAVE

PASS

BUT: Désigne et annule les mots de passe tuilisés pour limiter l'accès aux programmes.

REMARQUES: La commande PASS est utilisée pour permettre à l'ordinateur de ne fonctionner qu'en mode RUN afin de protéger les programmes contre les accès non autorisés. Un mot de passe se compose d'une chaîne de caractères délimitée par des guillemets de 8 caractères au maximum comportant toute combinaison des caractères de clavier plus les numéros 0 à 9, à l'exception du symbole''.

Une fois un mot de passe désigné, l'ORDINATEUR ne peut pas passer aux modes PRO ou RESERVE. Les opérations suivantes ne sont pas effectives et les lignes ne peuvent pas être ajoutées ou supprimées: LIST, LLIST, (C)SAVE, (C)LOAD, NEW, TITLE, MERGE, CHAIN et l'utilisation des touches ↑ et ↓.

S'il existe plusieurs programmes en mémoire lorsque le mot de passe est désigné, tous les programmes en mémoire sont protégés. Le seul moyen pour enlever la protection est d'exécuter une autre commande PASS avec le même mot de passe.

La commande ne peut pas être utilisée s'il n'existe aucun programme en mémoire. Pour changer le mot de passe, introduisez d'abord la commande avec l'ancien mot de passe, puis réintroduisez la commande avec le nouveau mot de passe.

EXEMPLE:

```
>PASS "SECRET"
```

Désigne le mot de passe "SECRET" pour tous les programmes en mémoire.

PAUSE

FORMAT: 1. PAUSE [<liste des expressions>] {[;] / [,]}

2. PAUSE USING <chaîne de format>;<liste des expressions>

Abréviation: PA.

Voir également: PRINT, WAIT

BUT: Affiche les données sur l'écran pendant une période de temps fixe.

REMARQUES: L'instruction PAUSE est identique à l'instruction PRINT sauf le fait que lorsqu'elle est utilisée en MODE 1, les données sont affichées sur l'écran pendant une période de temps fixe (0,85 seconde) et disparaissent de l'écran. Voyez la description de l'instruction PRINT. L'utilisation de la commande PAUSE est similaire à celle d'une commande WAIT précédant une commande PRINT.

En MODE 1, la <liste des expressions> peut contenir au maximum 2 articles si le séparateur est une virgule.

EXEMPLE:

```
10:PAUSE "LIRE LECI RAPIDEMENT"
20:PRINT "MAINTENANT, C'EST FINI"

>MODE 1
>RUN
```

[10] Imprimé le message qui est remplacé en moius de'une seconde.

PCONSOLE

FORMAT: 1. PCONSOLE "LPT1:",[<longueur de ligne>M<code EOL>],[<décalage>]
2. PCONSOLE "COMn:"\[<longueur de ligne>],[<code EOL>]

Abréviation: PCONS. *Voir également:*

BUT: 1. Désigne le format d'impression et le code de fin de ligne pour l'imprimante.
2. Désigne la longueur de ligne et le code de fin de ligne pour la communication à travers les ports série.

REMARQUES: 1. Dans le premier format, PCONSOLE "LPT1:" désigne le format pour l'imprimante connectée au port de l'imprimante actuellement en activité. Le paramètre <longueur de ligne> désigne la longueur de chaque ligne imprimée de façon à coïncider avec le nombre de colonnes imprimables par l'imprimante en cours d'utilisation. S'il est fixé à 0, la longueur de ligne n'est pas limitée. Autrement, il est fixé à une valeur comprise dans une gamme de 16 à 255. Le paramètre <code EOL> désigne le code de fin de ligne qui est émis à l'imprimante chaque fois qu'un code de retour chariot est rencontré en cours d'exécution de l'instruction LLIST, LPRINT ou LFILES de la manière suivante:

<Code EOL>	Code de fin de ligne
0	CR (&0D)
1	LF (&0A)
2	CR + LF

Toute autre valeur produira un code d'erreur. Le paramètre <décalage> désigne le décalage de page pour l'imprimante, c'est-à-dire le nombre de colonnes vierges imprimées en tête de chaque ligne. La désignation doit être égale ou inférieure à la valeur de <longueur de ligne> -4. Les valeurs par défaut sont toutes 0. (longueur de ligne infinie, CR, décalage nul).

NOTE:
Lorsque le nombre de caractères imprimés par LPRINT déplace le stylo de moins de 4 mm (c'est-à-dire lorsque les caractères sont très petits comme en cas de CSIZE1), le stylo ne revient pas à la marge de gauche après un changement de ligne. Afin d'éviter cet inconvénient, lors de l'impression dans cette situation, remplissez l'instruction d'impression d'espaces de sorte que le stylo soit déplacé d'au moins 4 mm lors de l'impression de E, G, LPRINT "A

2. Dans le second format, PCONSOLE "COMn:" désigne la longueur de ligne et le code de fin de ligne lors de l'émission d'un fichier par l'un des deux ports série.

COM1: Port série RS-232C.
COM2: Port série optique
COM: Port actuellement sélectionné par l'instruction SETDEV.

Le paramètre <longueur de ligne> désigne la longueur de chaque ligne du texte envoyée au port délimité par le code de fin de ligne tel que désigné dans le paramètre <code EOL>. S'il est fixé à 0, le texte envoyé au port n'est pas délimité en lignes. Autrement, il est fixé à une valeur comprise dans une gamme de 16 à 255.
Le paramètre <code EOL> désigne le code de fin de ligne qui est envoyé au port chaque fois qu'un code de changement de ligne est rencontré. Les valeurs désignées sont de 0,1 ou 2 dans le format 1. Lorsqu'un fichier est reçu à travers un accès séquentiel par la commande INPUT, le code de fin de ligne désigné ici sera interprété comme un simple retour chariot lorsqu'il est rencontré dans le fichier en cours de réception.

Les valeurs actuellement désignées pour tous les paramètres dans l'instruction PCONSOLE sont maintenues jusqu'à ce qu'elles soient changées par une autre instruction PCONSOLE.

Si la taille de caractère est changée en utilisant CSIZE, la longueur de ligne spécifiée par PCONSOLE changera dans la même proportion, mais le décalage restera inchangé.

EXEMPLE:

```
>PCONSOLE "LPT1I",42,2,3
```

Fixe la longueur maximum de ligne à imprimer à 42 caractères, et termine chaque ligne dans un retour chariot et un changement de ligne (CR + LF). Fixe la marge de gauche à la position 4, c'est-à-dire, les 3 premières colonnes sont vierges et l'impression commence à la colonne 4.

PEEK

FORMAT: 1. PEEK (<adresse>)
2. PEEK #(<bloc de mémoire>,<adresse>)

Abréviation: PE.
Voir également: POKE, XPEEK, XPOKE

BUT: Renvoie un octet de données depuis l'adresse spécifiée en mémoire.

REMARQUES Le paramètre <adresse> spécifie l'adresse en mémoire comprise dans une gamme de 0 à 65535 (&0 à &FFFF). Le paramètre <bloc de mémoire> spécifie un bloc de mémoire compris dans une gamme de 0 à 7 où l'octet de données requis est mémorisé. Dans le format 1, le bloc de mémoire est le bloc 0. Pour d'autres détails, référez-vous à la topographie mémoire dans les Annexes.

EXEMPLE:

```
>PEEK (0,100)
                                17
>
```

Renvoie les données à l'adresse en mémoire 100 (&64) dans le bloc de mémoire 0. La valeur en vigueur est 17.

PHONE ON/OFF/STOP

FORMAT: 1. PHONE ON
2.PHONE OFF
3.PHONE STOP

Abréviation: PH.
Voir également: ON PHONE GOSUB

BUT: Valide ou invalide les interruptions reçues à travers le port RS-232C via un modem téléphonique.

REMARQUES PHONE ON valide les interruptions à recevoir à travers le port RS-232C. L'instruction ON PHONE GOSUB peut alors être utilisée pour brancher l'exécution sur le sous-programme de traitement d'interruption concerné.

EXEMPLE: PHONE OFF invalide les interruptions provenant du port RS-232C. PHONE STOP invalide les interruptions provenant du port RS-232C, mais enregistre la demande d'interruption la plus récente de celles qui sont reçues. Si PHONE ON est exécutée par la suite, cette interruption est traitée immédiatement. La commande par défaut est PHONE STOP.

```
10:ON PHONE GOSUB 250
20:PHONE ON
:
40 PHONE OFF
```

PITCH

FORMAT: 1. PITCH [<pas de caractères>](,<interlignage>]

Abréviation: PI.
Voir également: CSIZE

BUT: Désigne le pas de caractères et l'interlignage pour l'imprimante en mode de texte.

REMARQUES Les interlignages désignés par la commande PITCH sont les suivants:

Paramètre <pas de caractères>	Pas de caractères réel
Valeur par défaut 4-240	6 x CSIZE <taille> x 0,2 mm <pas de caractères> x 0,2 mm

Paramètre <interlignage>	Interlignage réel
Valeur par défaut 4-255	12 x CSIZE <taille> x 0,2 mm <interlignage> x 0,2 mm

Les désignations PITCH sont remises aux valeurs par défaut après l'exécution de TEXT, GRAPH, CSIZE, LUST, TEST et PCONSOLE. La commande PITCH peut être utilisée en mode graphique, mais le paramètre <interlignage> est ignoré.

EXEMPLE:

```
10:PITCH 30,45
20:LPRINT PREMIERE LIGNE
30:LPRINT DEUXIEME LIGNE
40:LPRINT TROISIEME LIGNE
50:END
```

[10] Fixe le pas de caractères pour l'imprimante à 30 et l'interlignage à deux.
[20-40] Ces lignes seront sorties sur l'imprimante conformément à la spécification ci-dessus. Changez les paramètres de PITCH et voyez l'effet obtenu.

POINT

FORMAT: 1. POINT (X,Y) 2.
POINT (Xcol)

Abréviation: POI.
Voir également: GPRINT, PRESET, PSET

BUT: Renvoie la désignation de ia colonne à points ou à image binaire en position spécifiée sur l'écran

REMARQUES Dans le format 1, la fonction renvoie 1 ou 0 indiquant que le point en position spécifiée (X,Y) est désigné (1) ou n'est pas désigné (0). Les valeurs pour X sont normalement comprises dans une gamme de 0 à 155 (une ligne sur l'écran graphique) et les valeurs pour Y, dans une gamme de 0 à 31 (hauteur de l'écran).

Dans le format 2, la fonction renvoie un nombre décimal de 0 à 255 représentant la valeur de donnée d'image binaire dans la colonne à 8 points à l'emplacement horizontal spécifié (Xcol) sur la ligne en cours d'exécution de la manière suivante:

POINT (25) renverra la valeur en nombre décimal de donnée d'image à 8 bits dans la 25 ème colonne à points à partir de la gauche de l'écran dans la ligne en cours d'exécution. Si la valeur renvoyée est 98 par exemple, celle-ci se convertit en 01100010 en nombre binaire. La colonne à 8 points est alors:

	-0
X	-1
	-0
	-0
	-0
X	-1
X	-1
	-0

Référez-vous à la description de GPRINT en ce qui concerne les détails de représentation d'image binaire.

NOTE:

Bien que les valeurs de coordonnées X et Y doivent être comprises dans une gamme de 155 x 31 afin de renvoyer un point sur l'écran réel, elles peuvent être spécifiifiées dans la gamme allant de -32768 à 32767 sans produire d'erreur. Le nombre renvoyé dans ce cas est 0 pour les deux formats.

EXEMPLE:

```
300:N=POINT (77,15) 310:
IF N=l GOTO 500
:
:

:
:
```

[3001 Contrôle le point au milieu de l'écran à l'emplacement 77, 15 pour examiner s'il est en circuit (noir).

[310) Commande un branchement sur la ligne 500 si le point est en circuit. L'instruction POINT peut être utilisée de cette façon, afin de détecter les configurations sur l'écran et d'utiliser cette information pour faire un branchement du programme.

POKE

FORMAT: 1. POKE [#<bloc de mémoire>,]<adresse>,<liste des entiers>

Abréviation: PO.
Voir également: PEEK, XPEEK, XPOKE

BUT: Ecrit un octet ou une série d'octets de données en adresse(s) spécifiée(s) de mémoire.

REMARQUES

Le paramètre <adresse> spécifie l'adresse en mémoire compris dans une gamme de 0 à 65535 (&0 à &FFFF).

Le paramètre <bloc de mémoire> spécifié un bloc de mémoire compris dans une gamme de 0 à 7 où l'octet de donnée requis doit être mémorisé. Si aucun bloc de mémoire n'est spécifié, le bloc de mémoire par défaut est celui contenant l'étiquette début du fichier de programmes pour le programme en cours d'exécution.

Le paramètre <liste des nombres entiers> spécifie une liste d'un ou plusieurs octets de données sous forme de nombres entiers compris dans une gamme de 0 à 255 (&0 à &FF) séparés par des virgules, qui sont écrits dans les adresses consécutives en mémoire en partant de l'adresse spécifiée. Si la mémoire disponible est insuffisante pour contenir toutes les données dans la liste, un code d'erreur est produit. Pour les détails, se référer à la topographie mémoire dans les Annexes.

EXEMPLE:

```
>POKE #2,&FF00,255,255
>
```

En partant de l'adresse de mémoire &FF00 (65280) du bloc de mémoire 2, les deux valeurs, 255 et 255, sont introduites en mémoire. 255 est &FF. Ceci aura comme résultat de fixer 16 bits de mémoire à "1". Ne pas pratiquer l'utilisation de cette commande sans étudier la topographie mémoire et sans savoir où il faut écrire. Cet exemple n'étant qu'une démonstration, soulèvera des problèmes au niveau de l'exécution. Toutefois, une 'ALL RESET' permettra le rétablissement après la production des erreurs dues à un mauvais relevé de la mémoire

POWER

FORMAT: 1. POWER OFF
2. POWER AOFF [(n)]

Abréviation: POW.
Voir également: ALARMS, ARUN, WAKE$

BUT: Désigne la fonction de mise hors tension automatique et/ou permet de couper l'alimentation par l'intermédiaire d'un programme.

REMARQUES

POWER OFF met immédiatement l'ordinateur hors tension, par l'introduction directe ou par l'intermédiaire d'un programme.

En cas de POWER AOFF(h), si la valeur n est égale à zéro ou est omise, la mise hors tension automatique est fixée à 10 minutes après la dernière introduction par clavier ou la phase de traitement. Si la valeur n est égale à 1, la mise hors tension automatique est invalidée. La commande n'exercera aucun effet pour les valeurs autres que 0 ou 1. La désignation par défaut après une ALL RESET est la mise hors tension automatique après 10 minutes et la remise sous tension automatique par un modem téléphonique ou à l'heure spécifiée (voir la commande WAKE$).

EXEMPLE:

Désigne la mise hors tension automatique au bout de 10 minutes.

```
5:WAIT
10:PRINT "S'IL NY A PLUS DE TRAVAIL,JEME REPOSE"
20:POWER OFF


>POWER AOFF 0
>
```

PRESET

FORMAT: 1. PRESET (X,Y)

Abréviation: PRE.
Voir également: PSET, LINE

BUT: Remet le point aux valeurs de coordonnées spécifiées sur l'écran.

REMARQUES: La commande PRESET remet (met hors circuit) le point en position spécifiée par les valeurs de coordonnées (X,Y) comprises dans la gamme de l'écran graphique (155 x 31).

NOTE:
Bien que les valeurs de coordonnées X et Y doivent être comprises dans une gamme de 155 x 31 afin de remettre un point sur l'écran, elles peuvent être spécifiées dans la gamme allant de -32768 à 32767 sans produire une erreur.

EXEMPLE: Voir PSET.

FORMAT: 1. PRINT [<liste des expressions>]

PRINT

Abréviation: P.

Voir également: MODE, PRINT USING, WAIT

{[;] / [,]}

BUT: Fait sortir les données sur l'écran.

REMARQUES: Les positions dans lesquelles les articles de données de l'instruction PRINT sont affichés sur l'écran sont déterminées par le signe de ponctuation utilisé pour séparer les expressions dans la <liste des expressions>. Le point-virgule ou la virgule peut être utilisé, mais l'effet est différent. En MODE 1, la <liste des expressions> peut contenir au maximum 2 articles si le séparateur est une virgule.

BASIC divise d'écran d'affichage en zones composées chacune de 13 espaces. Lorsque les articles dans une liste d'impression sont délimités par des virgules, les articles sont affichés suivant ces zones fixées.

PRINT <liste des expressions> affiche sur l'écran les valeurs des expressions dans la <liste des expressions>. Les expressions dans la liste peuvent être des variables numériques ou alphanumériques, ou des chaînes réelles si elles sont délimitées par des guillemets. Le format exact d'affichage dépend des séparateurs utilisés entre les articles dans la <liste des expressions>. Si un point-virgule (;) est utilisé pour délimiter les articles dans la liste, les articles sont affichés immédiatement l'un après l'autre. Si une virgule (,) est utilisée, chaque article affiché est positionné en face d'une zone d'impression d'une longueur fixe de 13 espaces. Les données numériques et alphanumériques sont cadrées à gauche. Ceci signifie en effet que la donnée est affichée tous les 13 espaces, d'une façon équivalente aux tabulations posées toutes les 13 colonnes. En MODE 1, l'instruction PRINT arrête l'exécution et attend une ENTER avant de reprendre l'exécution du programme. En MODE 0, l'écran continuera à afficher tout ce qui apparaît sur l'écran pendant la durée définie par la dernière instruction WAIT avant la reprise de l'exécution. Voir la description de la commande WAIT en ce qui concerne les détails.

A la différence de l'instruction LPRINT, l'option TAB n'est pas disponible dans une instruction PRINT, mais les données peuvent être affichées dans les positions prédéterminées sur l'écran en utilisant l'instruction CURSOR, ou dans les formats spéciaux en utilisant l'option USING (voyez la description de PRINT USING).

Si la <liste des expressions> se termine par un point-virgule, la sortie suivante sur l'écran se continue sur la même ligne, tandis que si la liste se termine par sans un point-virgule, la sortie suivante est portée sur une nouvelle ligne.

L'utilisation de la commande PRINT toute seule fait sortir une ligne blanche sur l'écran; ceci est équivalent à un changement de ligne.

EXEMPLE:

```
10:A=1
20:B=22
30:C=333
40:PRINT
50:PRINT "LES VALEURS":A;B;C
60:PRINT "LES VALEURS",A,B,C
70:END

>RUN

LES VALEURS: 1  22  333
LES VALEURS:                1
          22                333
>
```

[40] PRINT seule imprime une ligne blanche.
[50] Utilise des point-virgules pour séparer les valeurs. Un seul espace précède l'impression de chaque valeur.
[60] Utilise des virgules pour séparer les valeurs. Les valeurs sont imprimées dans les zones d'impression fixes.

PRI1MT# / PRIIMT# USING

FORMAT: 1. PRINT#<numéro de fichier>,<liste des expressions>

2. PRINT#<numéro de fichier>,USING<chaîne de format>, <liste des expressions>
3. PRINT#["<nom de fichier>";]<liste des variables>

Abréviation: P.#/P.#U.

Voir également: INPUT#, OPEN, PRINT USING

BUT: Utilisé pour écrire les données dans un fichier séquentiel de sortie sur le disque souple, le module de disque RAM ou la bande cassette.

REMARQUES: 1&2. Disque souple ou module de disque RAM

#<numéro de fichier> est le numéro affecté au fichier lorsque celui-ci a été ouvert pour la sortie par l'instruction OPEN. Une tentative d'écriture sur un fichier qui n'a pas été ouvert produira un code d'erreur. En ce qui concerne la spécification de USING <chaîne de format>, réferez-vous à la description de l'instruction PRINT USING.

Si les articles dans la <liste des expressions> sont délimités par des points-virgules " ; ", les articles de données sont écrits sur le fichier sans aucun espace entre articles. Si les articles sont délimités par des virgules " , ", les articles de données sont écrits sur le fichier avec un espace entre articles. La virgule de séparation doit être écrite sur le fichier comme article de donnée lui-même, délimité par des guillements. Voyez le programme d'exemple dans la section relative à la Création d'un fichier dans le Chapitre 11.

3. Bande cassette

Lorsque le paramètre <"nom de fichier"> n'est pas spécifié pour la bande cassette, les données sont écrites à partir de la position actuelle de la tête de bande et le fichier n'est doté d'aucun nom. Dans le format 3, seule la virgule peut servir de séparateur dans la <liste des variables>.

La <liste des variables> peut contenir des variables numériques ou alphanumériques. Dans les deux cas, un soin doit être apporté à utiliser des séparateurs corrects lors de l'écriture sur le fichier, autrement des problème peuvent se présenter lorsque les données du fichier sont lues plus tard par l'instruction INPUT#. Les variables composées ne peuvent pas spécifier les éléments individuels; le tableau entier doit être spécifié sous forme A(*).

EXEMPLE: Voir OPEN et PRINT.

PRINT USING/USING

FORMAT: 1. PRINT USING <chaîne de format>;<liste des expressions> 2. USING <chaîne de format>

Abréviation: P.U./U.
Voir également: PRINT, LPRINT USING

BUT: Imprime les données numériques ou alphanumériques en utilisant le format spécifié dans le paramètre <chaîne de format>.

REMARQUES: La <chaîne de format> se compose des caractères spéciaux délimités par des guillements qui déterminent la taille et le format de la zone dans laquelle les expressions sont affichées. La <liste des expressions>-se compose des expressions numériques ou alphanumériques qui doivent être affichées. Chaque expression dans la <liste des expressions> doit être séparée de celle qui suit par un point-virgule.

Les caractères utilisés pour constituer la <chaîne de format> lorsque l'expression dans la <liste des expressions> est numérique, diffèrent de ceux lorsque l'expression est alphanumérique. Pour les expressions numériques, le signe # est utilisé pour constituer la chaîne de format, tandis que pour les expressions alphanumériques, le signe & est utilisé.

Il existe huit formats possibles indiqués dans les huit exemples suivants:

(1) `"###"`
```
 43
-57
```

Format en nombre entier: Un nombre entier de deux chiffres ou un nombre entier de deux chiffres avec un signe - (un signe + ne sera pas affiché).

(2) `"###."`
```
 98.
-43.
```

Format (en nombre entier) en virgule fixe:
Un nombre entier de deux chiffres (ou de deux chiffres et un signe —) suivi par la virgule décimale.

(3) `"###.##"`
```
 64.29
-13.44
```

Format en virgule fixe: Partie entière de deux chiffres (ou de deux chiffres et un signe -) plus partie fractionnaire de deux chiffres.

(4) "##.##^"
23,11E-05
−4.33E-02

Format en virgule flottante: Mantisse en virgule fixe plus le caractère exponentiel E- plus un exposant de deux chiffres (ou d'un chiffre et un signe -). Le caractère exponentiel plus l'exposant lui-même occuperont toujours 4 colonnes au minimum.

(5) "###,###."
34,567.
−22,444.

Format en nombre entier en unité de mine:
Un nombre entier de cinq chiffres (ou de cinq chiffres et un signe -) avec un point à la position de mille et une virgule décimale.

Format en nombre entier avec signe: Un nombre entier de trois chiffres précédé par un signe + ou -.

(6) "+###"
−988
+211

Format en nombre entier rempli: Un nombre entier de quatre chiffres au maximum (ou de quante chiffres et un signe -) dont les espaces éventuels en tête sont remplis d'astérisques, suivi par une virgule décimale. S'il est négatif, un signe - apparaîtra dans le premier espace, avant tout astérisque.

(7) "*####"
2345
**22
*230
−***1

Format en chaîne de caractères: Une chaîne de six caractères au maximum, cadrée à gauche.

(8) "&&&&&&"
ABCDEF
GH

Le nombre total des caractères de format numérique (# ou *) qui peuvent être placés dans une chaîne de format est de 11 dans les formats (1), (2), (3), (4), (6) et (7), et est de 14 dans les formats (5). PC-1600 fait sortir les données sous forme de 10 chiffres significatifs si aucune chaîne de format n'est spécifiée (voir Section 4.1.4 "Types de données"). L'instruction USING peut être utilisée toute seule. Sous cette forme, le paramètre <chaîne de format> spécifié dans l'instruction USING reste valide pour toutes les instructions PRINT subséquentes jusqu'à rencontrer une autre instruction USING.

Pour effacer le paramètre <chaîne de format> en vigueur, exécuter une instruction PRINT USING ou USING sans spécifier aucune chaîne de format.

EXEMPLE:

```
10:A=1
20:B=22
30:C=33
40:USING "####.##"
50:PRINT "LES VALEURS SONT"
60:PRINT A;B;C
70:END

>RUN
LES VALEURS SONT
   1.00  22.00  33.00
>
```

[40] Définit le format pour l'impression des données numériques; deux chiffres plus un signe avant la virgule décimale et deux chiffres après celle-ci.
[50] Fait sortir sur l'écran les valeurs conformément au format spécifié.

PSET

FORMAT: 1. PSET (X,Y) [,<code fonctionnel]

Abréviation: PS.
Voir également: PRESET, UNE

BUT: Met ou remet le point aux valeurs de coordonnées spécifiées sur l'écran.

REMARQUES : La commande PSET positionne (met en circuit) le point aux valeurs de coordonnées spécifiées par (X,Y) dans la gamme de l'écran graphique (155 x 31). L'option <code fonctionnel> est spécifiée sous forme d'une majuscule X. Une fois PSET spécifiée, elle inverse l'état aux valeurs de coordonnées spécifiées. Si le point est en circuit, il est mis hors circuit et vice versa.

NOTE:
Bien que les valeurs de coordonnées X et Y doivent être comprises dans une gamme de 155 x 31 pour positionner un point sur l'écran réel, elles peuvent être spécifiées dans la gamme allant de -32768 à 32767 sans produire une erreur.

EXEMPLE:

```
10:FOR Y=10 TO 20 STEP 2
20:FOR X=10 TO 100 STEP 10
30:PSET (X,Y)
40:NEXT X
50:NEXT Y
60:FOR Y=10 TO 20 STEP 2
70:FOR X=10 TO 100 STEP 10
80:PRESET (X,Y)
90:NEXT X
100:NEXT Y
110:END
```

Ceci tracera une grille ou des points à travers l'écran, puis les supprimera.

[10] Désigne la boucle pour les rangées.
[20] Désigne la boucle pour les colonnes.
[30] Chaque fois que cette instruction est exécutée, un seul point sur l'écran est mis en circuit (noir).
[60-100] Répète le processus des lignes 10-50 mettant chaque point hors circuit.

FORMAT: 1. PZONE "COMn:",<largeur> 2.
PZONE "LPT1:",<largeur>

PZONE

Abréviation: PZ.
Voir également: LPRINT

BUT: Désigne la zone d'impression lors de la sortie sur l'imprimante ou sur un port série par l'instruction LPRINT.

REMARQUES, Dans BASIC, l'impression par l'instruction LPRINT est divisée en zones composées chacune d'un nombre fixe d'espaces. Les articles imprimés par la commande LPRINT en utilisant la virgule comme séparateur sont imprimés dans ces zones en cadrant à droite les articles numériques et les données alphanumériques dans chaque zone. L'instruction PZONE spécifie la largeur de ces zones.

Les données peuvent également être envoyées à un port série par l'instruction LPRINT. Dans ce cas, PZONE spécifie le découpage en zones des articles envoyés au port de la manière exactement la même que pour l'imprimante.

L'un des ports série suivants peut être spécifié:

COM1: Port série RS-232C
COM2: Port série optique
COM: Port actuellement sélectionné par la dernière instruction SETDEV

Pour C0M1 :, la gamme de <largeur> est de 8 à 255. Pour LPT1 :, la gamme de <largeur> est de 8 à 80.

La désignation par défaut pour le paramètre <largeur> lors de la mise sous tension est 20.

EXEMPLE:

```
10:A=1
20:B=22
30:C=333
40:PZONE "LPT1:",10
50:LPRINT A,B,C
60:END

>RUN
           1          22         333        (Sur l'imprimante)
>
```

[40] Fixe les largeurs de zone pour l'imprimante à 10.
[50] Fait sortir sur l'imprimante les trois valeurs dans les colonnes spécifiées par PZONE.

RADIAN

FORMAT: 1. RADIAN

Abréviation: RAD.
Voir également: DEGREE, G RAD

BUT: Met l'ordinateur en mode radians.

REMARQUES: En mode radians, RADIAN est affiché sur la ligne supérieure de l'écran. L'introduction des arguments de SIN, COS et TAN doit être effectuée en radians. Les valeurs renvoyées par les fonctions ASN, ACS et ATN seront en radians.

EXEMPLE:

```
10:RADIAN
20:PRINT "LES FONCTIONS TRIG SONT MAINTENANT EN RADIANS"
30:PRINT ASN(0.5),ASN(1.0)
40:PRINT ACS(0.5),ACS(1.0)
50:PRINT ATN(0.5),ATN(1.0)
60:END
```

Exécutez ce programme et comparez les résultats avec ceux pour DEGREE et GRAD.

(10) Sélectionne le mode RADIAN.
[30-50] Imprime quelques valeurs d'exemple.

RANDOM

FORMAT: 1. RANDOM

Abréviation: RA. *Voir également:* RND

BUT: Change la série des nombres aléatoires à produire par la fonction RND.

REMARQUES: L'utilisation de RANDOM au début d'un programme change la séquence de production des nombres aléatoires chaque fois que l'ordinateur est mis sous tension.

EXEMPLE:

```
>RANDOM
>RUN
```

Change la série des nombres aléatoires qui seront produits par RND dans le programme avant l'exécution de ce dernier.

RCVSTAT

FORMAT: 1. RCVSTAT "COMn:",[<protocole>],[<temporisation>]

Abréviation: RC.

Voir également: INSTAT, SNDSTAT

BUT: Désigne le protocole de connexion de réception et la valeur de temporisation pour le port RS-232C.

REMARQUES:

L'un des ports série suivants peut être spécifié:

COM1 : port série RS-232C
COM2: port série optique
COM: port actuellement sélectionné par la dernière instruction SETDEV

La commande RCVSTAT établit les conditions de signal pour la réception de données à travers le port RS-232C. Si les conditions désignées sont satisfaites, les données peuvent être reçues à travers le port. Ceci s'appelle la "connexion". Les états de signaux auxquels les données seront reçues sont désignés par la valeur du nombre décimal spécifié pour le paramètre <protocole>. Ce nombre est intérieurement converti en nombre binaire à 8 bits et chaque bit désigne la condition d'un signal de commande pour effectuer la connexion. Si toutes les conditions sont satisfaites, la connexion est terminé et la réception de données peut être effectuée.

Bit	Valeur	Condition de signal
1 (LSB)	0	DTR doit être haut.
	1	DTR doit être bas.
2	0	RTS doit être haut.
	1	RTS doit être bas.
3	0	CTS doit être haut.
	1	CTS doit être bas.
4	0	CD doit être haut.
	1	CD doit être bas.
5	0	DSR doit être haut.
	1	DSR doit être bas.
6	0	CI doit être haut.
	1	CI doit être bas.
7	Non utilisé; fixé à 0.	
8 (MSB)	Non utilisé; fixé à 0.	

La désignation par défaut pour le paramètre <protocole> est 63 (00111111), qui permettra la réception de données sans aucun signal de commande. L'option protocole n à aucune signification pour le port série optique.

La temporisation est la durée pendant laquelle l'ordinateur attend l'introduction sur un port jusqu'à décider que l'unité faisant l'objet de l'attente n'est pas prêteou n'est pas disponible. Le paramètre <temporisation> optionnel désigne la temporisation dans une gamme de 0 à 255, à des échelons de 0,5 secondes. Une temporisation de 0 invalide la désignation de temporisation (temporisation infinie) et est la valeur par défaut.

EXEMPLE:

```
>RCVSTAT"COM1:",45
```

Désigne les conditions de signal pour le port série RS-232C spécifiant que les états de RTS et de DSR doivent être hauts pour permettre la réception de données. La valeur de temporisation est infinie. Le nombre 45 en valeur décimale correspond à 2D en valeur hexadécimale et représente la configuration
binaire suivante:

Notez que le bit le plus significatif ou de poids fort MSB est le bit 8, tandis que le bit le moins significatif ou de poids faible LSB est le bit 1.

READ .. DATA

FORMAT: 1. READ <liste des variables>

DATA <liste des constantes>

Abréviation: READ. DA.
Voir également: DATA, RESTORE

BUT: Lit les valeurs d'une instruction DATA et les affecte aux variables.

REMARQUES: L'instruction READ doit toujours être accompagnée d'au moins une instruction DATA. Chaque article dans la <liste des constantes> est affecté à chaque article dans la <liste des variab!es> sur une base d'article à article, bien qu'une seule instruction READ ne corresponde pas nécessairement à une seule instruction DATA.

Les articles dans la <liste des constantes> sont séparés par des virgules. Le type de variable doit coïncider avec le type de constante pour la lecture de chaque article. Voir l'instruction DATA. Une fois les articles de DATA lus, ils ne peuvent pas être relus à moins qu'une instruction RESTORE ne soit exécutée.

Si le nombre de variables dans la <liste des variables> est supérieur au nombre d'articles dans la<liste des constantes>, un code d'erreur se produit. Si le nombre de constantes est supérieur au nombre de variables, les constantes en excès sur la
ligne de données resteront non lues.

Référez-vous à la commande MERGE en ce qui concerne les détails d'utilisation des commandes READ et DATA lors de l'interclassement des programmes.

EXEMPLE:

```
5:WAIT 40
10:DIM B(10)
20:FOR I=1 TO 10
30:READ B(I)
40:PRINT B(I)
50:NEXT I
60:DATA 10,20,30,40,50
70:DATA 60,70,80,90,100
80:END

>RUN
10
20
30
40
50
60
70
80
90
100
>
```

[5] Spécifie le teur d'attente pour I affichange de l'écran. [10] Réserve l'espace pour un tableau à une dimension. [20] Lit les valeurs des instructions DATA, B(1) est 10, B(2) est 20, B{3) est 30, etc., et imprime les
valeurs.

REM

FORMAT: 1. REM «digne de remarque> 2.
' <ligne de remarque>

Voir également: LIST, LLIST

BUT: Utilisée pour inclure les commentaires dans un programme.

REMARQUES Une apostrophe (') BUILU + Q peut être utilisée à la place du mot-clé REM. Les lignes de remarques sont ignorées par BASIC durant l'exécution du programme, mais apparaissent exactement comme elles sont écrites lorsque le programme est LISTé. Les commentaires sur ce que le programme effectue sont extrêmement utiles pour d'autres personnes qui peuvent désirer utiliser ou modifier le programme. Si une GOTO, ou d'autres instructions de branchement similaires, réalise un branchement sur une ligne de REM, l'exécution recommencera par la première ligne subséquente contenant une instruction exécutable. Les remarques peuvent être ajoutées aux lignes contenant déjà des instructions exécutables en utilisant des deux-points et ensuite REM, suivie par la remarque. Aucune autre instruction ne peut être placée après l'instruction REM sur la même ligne. L'apostrophe (') ne peut pas être utilisée dans la forme d'addition.

EXEMPLE:

```
10:REM CETTE LIGNE N'A AUCUN EFFET
20:DIM A$(3,5):REM ESPASE RESERVE POUR UN TABLEAU 3×5
30:CETTE LIGNE  N'A EGALEMENT AUCUN EFFET
```

RENUM

FORMAT: 1. RENUM [<nouveau numéro de ligne>],[<ancien numéro de ligne>],
[<incrément>]

Abréviation: REN.
Voir également: DELETE, EDIT, LIST

BUT: Renumérote les lignes d'un programme.

REMARQUES: Les numéros de lignes sont changés de l'oncien numéro de ligne> au <nouveau numéro de ligne> avec l'<incrément> spécifié. Si le <nouveau numéro de ligne> n'est pas spécifié, les lignes sont renumérotées en partant de 10 par incréments de 10. Tous les numéros de ligne cités dans GOTO, GOSUB et d'autres instructions de branchement sont changés par la suite. Lorsque le numéro de ligne cité dans une instruction GOTO ou GOSUB ne peut pas être trouvé dans le programme en cours de renumérotation, le programme n'est pas renuméroté et un message "non défini dans le <numéro de ligne>" est affiché. Si une instruction GOTO indique une expression au lieu d'un numéro de ligne, la renumérotation n'est pas effectuée. Le plus haut numéro pouvant être affecté est 65279.

EXEMPLE:

```
10:INPUT "CONTINUE";A$
20:IF A$="OUI"THEN 10
30:IF A$="NON"THEN 60
40:PRINT "PRIERE D'ENTRER OUI OU NON"
50:GOTO 10
60:END

>RENUM 100,10,5
>LIST
100:INPUT "CONTINUE";A$
105:IF A$ "OUI"THEN 100
100:IF A$ "OUI"THEN 25
115:PRINT "PRIERE D'ENTRER OUI OU NON"
120:GOTO 100
125:END
>
```

RESTORE

FORMAT: 1. RESTORE
2. RESTORE <numéro de ligne/étiquette>

Abréviation: RES.
Voir également: READ, DATA

BUT: Permet la relecture des instructions DATA en partant de la ligne spécifiée.

REMARQUES Normalement, l'instruction READ lit les articles des lignes DATA à partir du premier article sur la ligne DATA portant le numéro le plus bas. Elle continue ce processus en succession pour chaque instruction READ subséquente dans le programme. La position en cours de lecture est retenue par un "indicateur" stocké en mémoire..RESTORE remet l'indicateur au premier article dans la première ligne de données dans le programme. RESTORE <numéro de ligne/étiquette> remet l'indicateur au premier article du numéro de ligne spécifié. Si le numéro de ligne spécifié ne contient pas une instruction DATA, l'indicateur est remis au premier article de données sur la première ligne de données qui suit le numéro de ligne spécifié.

EXEMPLE:

```
10:DIM A1$(3),A2$(3)
20:FOR N=1 TO 3
30:READ A1$(N)
40:PRINT A1$(N)
50:NEXT N
60:RESTORE
70:FOR N=1 TO 3
80:READ A2$(N)
90:PRINT A2$(N)
100:NEXT N
110:END
120:DATA "PRUNE","PECHE","BRUGNON"
```

```
> RUN
PRUNE
PECHE
```

```
BRUGNON

PRUNE
PECHE
BRUGNON
>
```

[10] Détermine les dimensions des deux tableaux de chaînes.

[20-50] Lit les trois chaînes de données, les affecte dans le tableau à trois éléments A1$ et les imprime.

[60] Rétablit toutes les instructions de données de sorte qu'elles puissent être relues.

[70-100] Lit les mêmes chaînes de données, les affecte dans le second tableau à trois éléments A2$ et le imprime. Les deux tableaux contiennent maintenant les données identiques.

RESUME

FORMAT: 1. RESUME [<numéro de ligne>] 2.
RESUME NEXT

Abréviation: RESU.
Voir également: ON ERROR GOTO, ERL, ERN

BUT: Utilisée pour reprendre l'exécution normale d'un programme après qu'en présence d'une erreur, l'exécution a fait un branchement sur le sous-programme de traitement d'erreurs d'utilisateur.

REMARQUES: La commande RESUME exerce le même effet sur un sous-programme de traitement d'erreurs que l'effet exercé par l'instruction RETURN sur un sous-programme.

RESUME <numéro de ligne> fait reprendre l'exécution de l'instruction ayant le numéro de ligne spécifié. RESUME fait reprendre l'exécution de l'instruction qui a causé l'erreur. RESUME NEXT fait reprendre l'exécution de la ligne qui suit la ligne contenant l'erreur. L'instruction RESUME ne peut pas être utilisée n'importe où dans un programme sauf dans un sous-programme de traitement d'erreurs.

EXEMPLE: Voir ON ERROR GOTO.

RETI

FORMAT: 1. RETI

Abréviation:

Voir également: ON ADIN GOSUB, ON COMn GOSUB, ON KEY GOSUB, ON PHONE GOSUB, ON TIMES GOSUB

BUT: Rétablit l'exécution de programme à la place d'un sous-programme d'interruption.

REMARQUES: RETI est utilisée de la même manière que RETURN pour renvoyer à une ligne suivante, la ligne qui a spécifié un appel de sous-programme. La différence est que RETI est utilisée avec des sous-programmes lancés par les interruptions: ON ADIN GOSUB, ON COMn GOSUB, ON KEY GOSUB, ON PHONE GOSUB et ON TIME$ GOSUB. Lorsque RETI est exécutée, la dernière interruption est exécutée dans le cadre du programme d'exécution d'interruption.

EXEMPLE:

```
10:ON PHONE GOSUB 100
:
:
100:PRINT "VOUS AVEZ UN APPEL"
110:RETI
```

RIGHT$

FORMAT: 1. RIGHT$(X$,N)
2. RIGHT$("<chaîne>",N)

Abréviation: RI.
Voir également: LEFT$, MID$

BUT: Renvoie les caractères N de l'extrémité droite d'une chaîne X$ quelconque.

REMARQUES: La valeur de N doit être comprise entre 0 et 80. Les fractions seront arrondies à l'unité inférieure (tronquées). Si N est inférieur à 1, une chaîne vide est renvoyée. Si N est supérieur au nombre de caractères dans X$, Ja chaîne entière est renvoyée.

EXEMPLE:

```
5:WAIT 32
10:X$="SHARP PC1600"
20:FOR N=1 TO 13
30:LET S$=RIGHT$ (X$,N)
40:PRINT S$
50:NEXT N

>RUN
0
00
600
1600
C1600
PC1600
 PC1600
P PC1600
RP PC1600
ARP PC1600
HARP PC1600
SHARP PC1600
SHARP PC1600
>
```

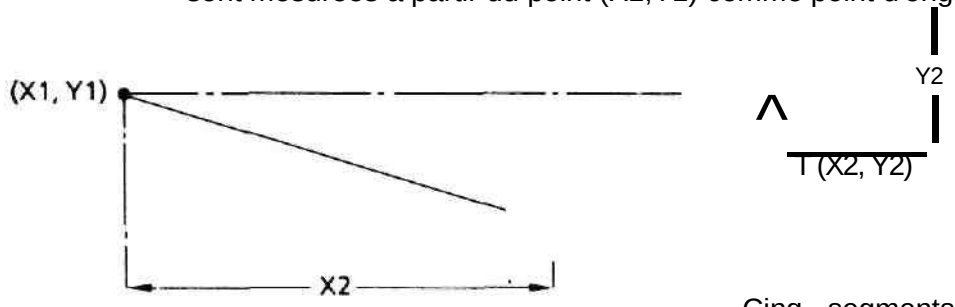
RLINE

FORMAT: 1. RLINE [(X1,Y1)]-(X2,Y2) [-(X3,Y3)..],[<type>],[<couleur>][,B]

Abréviation: RL.
Voir également: COLOR, LLINE

BUT: Trace un trait ou une série de segments de trait entre les points spécifiés dans les coordonnées relatives sur l'imprimante.

REMARQUES: En mode de graphique, la commande RLINE trace un segment de trait à partir du point spécifié par les coordonnées X1, Y1 ou à partir de la position actuelle du stylo si X1 et Y1 ne sont pas spécifiées, jusqu'au point spécifié par les coordonnées relatives X2, Y2. A la différence de la commande LLINE, les coordonnées dans la commande RLINE sont toutes relatives par rapport à la dernière position du stylo en tant que point d'origine. C'est-à-dire que les coordonnées (X2,Y2) sont mesurées à partir du point (X1,Y1) comme point d'origine.



Cinq segments de trait contigus subséquents au maximum peuvent être spécifiés avec les coordonnées de chaque point neuf en cours de mesure à partir du dernier point comme point d'origine.

Le paramètre optionnel <type> spécifie le type de trait dans une gamme de 0 à 9. (Voir la commande LLINE.) le paramètre optionnel <couleur> spécifie la couleur du stylo dans une gamme de 0 à 3 (Voir la commande COLOR). Si les paramètres <type> et <couleur> ne sont pas spécifiés, les valeurs actuellement désignées seront utilisées. L'option B spécifie un rectangle à tracer sur la diagonale spécifiée par XI, Y1 et X2, Y2 comme suit:

Pour la commande RLINE (X1,Y1)-(X2,Y2),„B

EXEMPLE:

```
5IPAPER R
10:GRAPH
20:GLCURSOR (40,40)
30:RLINE -(100,0)-(~100,100)-(0,-100)
40ITEXT
50:END
```

Trace un triangle éloigné du coin de la page. Le programme aboutit au même résultat que l'exemple pour LLINE.

[10] Met l'unité en mode de graphique.
(20] Déplace le stylo d'imprimante à la position spécifiée éloignée du coin de la page.
[30] Trace le triangle par tois mouvements et renvoie le stylo à la position initiale. Noter qu'à la différence de l'exemple de LLINE, tous les mouvements de la plume pour RLINE sont relatifs par rapport à là dernière position du stylo.
(40] Une fois le traçage terminé, rétablir le mode de texte.

RMT ON/OFF

FORMAT: 1. RMT ON
2. RMT OFF

Abréviation: RM.
Vb/r *également:*

BUT: Valide et invalide la télécommande d'alimentation de l'enregistreur à cassette.

REMARQUES: RMT ON permet à l'enregistreur à cassette d'être mis sous tension par le signal RMT provenant de l'ordinateur durant une opération E/S de bande et autrement d'être mis hors tension.

RMT OFF invalide la télécommande d'alimentation de l'enregistreur à cassette; le courant restera indépendamment de l'opération E/S de bande

La fonction de télécommande est possible seulement lorsque l'interrupteur REMOTE ON/OFF sur l'unité CE-1600P est enclenché.

EXEMPLE:

```
>RMT  ON
```

Valide la commande par ordinateur de l'enregistreur à cassette.

```
>RMT  OFF
```

Invalide la commande par ordinateur de l'enregistreur à cassette.

RIMD

FORMAT: 1. RND(X)

Abréviation: RN.
Voir également: RANDOM

BUT: Renvoie un nombre aléatoire dans une gamme spécifiée par le nombre X.

REMARQUES: L'ordinateur engendre un nombre aléatoire à l'aide d'un générateur de nombres aléatoires incorporé. La même séquence de nombres aléatoires est engendrée chaque fois qu'un programme contenant RND est exécuté après la mise sous tension.

EXEMPLE: Si X est inférieur à 0, le générateur de nombres aléatoires répète la même séquence de nombres aléatoires chaque fois que RND est exécutée. Si X est une valeur décimale supérieure à 0 et inférieure à 1, le nombre aléatoire est engendré dans une gamme de 0 à 1 (1 non compris), Si X est supérieur à 1, le nombre aléatoire est engendré dans une gamme de 1 à X en nombres entiers.

Tous les nombres aléatoires sont engendrés en 10 chiffres significatifs.

```
10:FOR I=1 TO 3
20:FOR J=1 TO 10
30:R=RND (9)
40:PRINT R;
50:NEXT J
60:PRINT
70:NEXT I
80:END
```

```
>RUN
6  4  2  5  6  8  2  7  6  8
5  5  7  7  1  2  6  5  3  6
3  1  5  7  3  4  5  7  4  2
>

>RUN
2  9  1  1  2  3  3  3  3  5
8  7  4  5  7  5  8  1  4  8
2  8  2  5  9  1  9  4  7  6
>
```

[10] Boucle externe pour chaque ligne de nombres aléatoires.
[20] Boucle interne pour génération de 10 nombres aléatoires.
[30] Engendre un nombre aléatoire compris entre 1 er 9.

Ici, le programme est passé trois fois. Noter qu'une séquence différente de nombres aléatoires est engendrée à chaque fois.

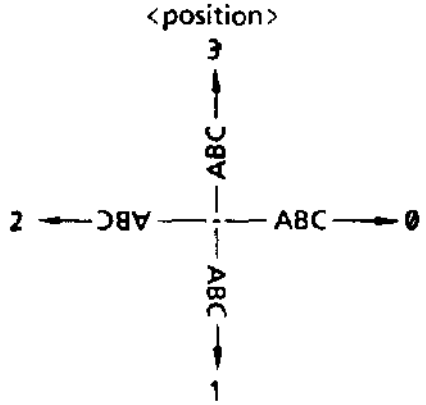
ROTATE

FORMAT: 1. ROTATE <position>

Abréviation: RO. *Voir également:*

BUT: Spécifie l'orientation de l'impression de chaque caractère.

REMARQUES: Le paramètre <position> spécifie l'orientation des caractères à imprimer sous forme d'un nombre de 0 à 3. L'effet du paramètre <position> lors de l'impression est donné sur les deux illustrations suivantes:



Les valeurs par défaut sont celles spécifiées par la dernière commande ROTAtE.

EXEMPLE:

```
5:GRAPH
10:GLCURSOR (400,0)
20:FOR P=0 TO 3
30:ROTATE P
40:LPRINT "ABC ";
50:NEXT P
60:END
```

Le passage de ce programme démontrera toutes les 16 combinaisons possibles pour valeurs d'impression ROTATE.

[5] Spécifie le mode graphique.

[10] Déplace la tête d'imprimante au milieu du papier.

[30] Tourne sur une boucle par les 4 positions possibles de caractère.

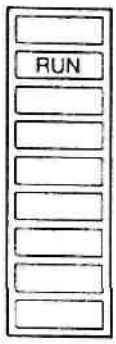
[40] ROTATE (oriente) suivant les valeurs de boucle.

[50] Fait sortir la chaîne simple sur imprimante.

[60-70] Répète la boucle.

FORMAT: 1. RUN
2. RUN <Numéro de ligne/étiquette>

RU
N



Abréviation: R.
Voir également: CONT, GOTO, LOAD, MERGE

BUT: Lance l'exécution d un programme tenu en mémoire.

REMARQUES: La commande RUN lance l'exécution du programme en mémoire à partir de l'instruction du numéro le plus jeune. RUN <n° de ligne/ étiquette> lance l'exécution à partir du numéro de ligne spécifié ou de l'étiquette spécifiée. L'exécution d'un programme par l'instruction RUN efface toutes les va-rialbles, range et remet le pointeur de données sur une ligne DATA (données) pour le lancement.

EXEMPLE:

```
>RUM
```

Lance l'exécution d'un programme à partir de la linge d'un numéro le plus jeune.

```
>RUN  100
```

Lance l'exécution d'un programme à partir de la ligne 100.

```
>RUN "F"
```

Lance l'exécution d'un programme à partir d'une ligne avec l'étiquette "F".

RXD$

FORMAT: 1. RXD$

Voir également.

BUT: Renvoie les données introduites dans le port série actuellement choisi, lors de l'exécution.

REMARQUES: L'octet de données introduites dans le port actuellement choisi par l'instruction SETDEV est renvoyé sous forme d'une chaîne hexadécimale. S'il n'y a pas de données actuellement introduites, deux octets de données, &20 + &20 sont renvoyés. S'il y a une erreur de communication lors de l'exécution, trois octets de données, &3F + &20 + &20 sont renvoyés.

EXEMPLE:

```
>PRINT RXD$
```

Fait sortir l'octet de données actuellement introduites au port série actif,

SAVE/SAVE*

FORMAT: 1. SAVE "<d:nom de fichier>"[,A] 2.
SAVE* "<d:nom de fichier>"

Abréviation: S.
Voir également: LOAD, MERGE

BUT: Utilisée pour conserver le programme ou le fichier de données sur le disque souple ou module de disque RAM, bande, ou l'un des ports série.

REMARQUES: SAVE conserve le fichier sur une unité spécifiée sous le nom de fichier spécifié. Si l'option A est spécifié, le fichier est conservé dans un format ASCII, autrement il est conservé dans un format binaire comprimé. <d:nonn de fichier> spécifie l'unité et le nom de fichier.

SAVE* conserve seulement les lignes comentaires (préfixées par REM ou une apostrophe ('); voir l'instruction REM) sous le nom de fichier spécifié. D'autres lignes sont ignorées. Cela donne une méthode simple de stockage du texte dans un fichier pour un affichage ultérieur sur l'écran. Pour les détails concernant le stockage et l'extraction du texte, voir la description de l'instruction LOAD*.

Si aucune extension n'est spécifiée pour le nom de fichier, SAVE connectera automatiquement .BAS. d'extension.

Si le nom de fichier spécifié existe déjà, le fichier existant sera recouvert, à moins qu'il ne soit protégé par l'option P de la commande SET ou que le disque ou le module ne soit protégé en écriture.

EXEMPLE:

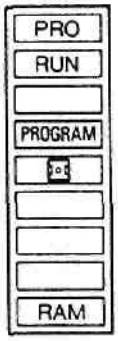
```
>SAVE"X:ECHECS"
```

Conserve le programme en mémoire sur le disque souple sous le nom de ECHECS.

```
>SAVE*"S2:iNFO"
```

Conserve simplement les lignes commentaires du programme en mémoire sur le module de disque RAM dans le logement d'extension 2. Le fichier est appelé INFO.

SET



FORMAT: 1. SET "<d:nom de fichier>","P" 2.
SET "<d:nom de fichier>"," "

Abréviation: Voir également:

BUT Spécifie et supprime la protection de fichier pour les fichiers sur le disque souple ou le disque RAM.

REMARQUES: La commande SET est utilisée pour spécifier les paramètres afin de protéger un fichier contre un port non admissible ou une élimination accidentelle. Le paramètre <d:nom de fichier> spécifie l'unité et le nom de fichier.

Une fois l'option P, protection en écriture, spécifiée, le fichier est protégé comme suit:

(1)Aucune donnée ne peut être écrite sur le fichier. Le fichier ne peut être ouvert eh mode APPEND ou OUTPUT.
(2)Le fichier ne peut être effacé par la commande KILL.
(3)Il n'est pas possible de donner au fichier un nouveau nom par la commande NAME.

Pour annuler l'état de protection P sur un fichier, exécutez la commande SET pour le nom du fichier avec un espace à la place du paramètre "P" (format 2).

EXEMPLE:

```
>SET"X:SALAIRES","P"
>
```

Protège le programme SALAIRES tenu sur le disque souple contre l'écriture, l'effacement ou le changement de nom.

SETCOM

FORMAT: 1. SETCOM "COMn:",[
],[<WL>],[<PR>],[<ST>],[<XO>],[<SI>]

Abréviation: SETC.
Voir également: COM$, SETDEV

BUT: Spécifie le protocole de transmission pour les port série.

REMARQUES: L'un des ports série suivants peut être spécifié:

COM1: Port série RS-232C
COM2: Port série optique
COM: Port actuellement choisi par la dernière instruction SETDEV

Le protocole de transmission est spécifié sous forme d'une chaîne de six options, séparées par virgules. Les options sont:

 Débit: 50 à 38400 bauds <WL>
Longueur de mot: 5 à 8 bits <PR>
Parité: E-Parité paire
O-Parité impaire N-Sans parité <ST> Bit d'arrêt: 1 ou 2 <XO> Protocole X-ON/OFF: X-Oui
N-Non
<SI> Protocole code normal/spécial
Pour 7 bits utiles seulement: S-Oui
N-Non

La consigne par défaut pour le port série RS-232C est 1200,8,N,1,X,S. La valeur par défaut pour le port série optique est 38400,7,E,2,X,S.

NOTE:
Lorsque les commandes SAVE, LOAD, BSAVE ou BLOAD sont utilisées pour tranférer des fichiers par un port, la longueur de mot doit être spécifiée à 8 bits, et le protocole de code normal/spécial doit être spécifié à N. Cependant, lors de la conservation d'un fichier ASCII dans un port par SAVE, ou lors de l'utilisation PRINT# ou INPUT# avec un port, la spécification des paramètres de communication (ou de transmission) est illimitée.

EXEMPLE:

```
>PRINT COM$ "COM1:"
300,7,N,1,N,N
>SETCOM "COM1:",,,E
>PRINT COM$ "COM1:"
300,7,E,1,N,N
>
```

La première ligne renvoie la valeur de communication actuelle du port série RS-232C, SETCOM est utilisée pour changer la valeur de parité de "sans parité" à "parité". PRINT COM3 est alors introduite au clavier pour vérifier que le changement a été effectué. Notez que le troisième paramètre est maintenant E au lieu de N.

SETDEV

FORMAT: 1. SETDEV "COMn:" [,PO][,KI]

Abréviation: SE. *Voir également:*

BUT: Spécifie un des ports série pour entrée et sortie avec certaines commandes BASIC.

REMARQUES: La commande SETDEV ouvre fondamentalement un accès pour entrée et sortie. De plus, deux paramètres optionnels peuvent être spécifiés pour définir le mode d'entrée ou de sortie. Les accès sont spécifiés comme suit:

COM1 : Port série RS-232C COM2: Port série optique COM: Port actuellement ouvert.

Les deux paramètres optionnels sont spécifiés comme suit (dans n'importe quel ordre):

PO	Dirige la sortie depuis les commandes LPRINT, LLIST et LFILES au à port spécifié.
KI	Utilise le port spécifié po.ur introduction de données par une commande INPUT.

La spécification de la commande SETDEV sans aucun paramètre annule les consignes en cours de l'unité et permet à nouveau une sortie sur imprimante et une introduction au clavier.

EXEMPLE:

```
>SETDEV "COM1:",PO

>SETDEV
```

SGN

FORMAT: 1. SGN(X)

Abréviation: SG. *Voir également:*

BUT: Renvoie le signe d'expression X.

REMARQUES : X est une expression numérique quelconque. SGN(X) renvoie 1,0 ou -1 selon le signe d'expression X comme indiqué ci-dessous:

Valeur de X	Valeur renvoyée par SGN(X)
X>0	1
X=0	0
X<0	-1

EXEMPLE:

```
5:WAIT 100
10:FOR N=-3 TO 3
20:PRINT N,SGN (N)
30:NEXT N
40:END


>RUN
              -3              -1
              -2              -1
              -1              -1
               0               0
               1               1
               2               1
               3               1
>
```

SIN

FORMAT: 1. SIN(X)

aviation: SI. *Voir également:* ASN, COS, TAN

BUT: Renvoie le sinus de l'angle X.

REMARQUES: Cette fonction renvoie le sinus de l'angle X, où X est exprimé en degrés, radians ou grandients. Selon cette expression, le mode de l'ordinateur est spécifié par la commande DEGREE, RADIAN ou GRAD.

EXEMPLE:

```
10:DEGREE
20:PRINT "LE SIN DE 30 EST ";SIN (30)
30:PRINT "LE SIN DE 90 EST ";SIN (90)
40:END

>RUN
SIN OF 30 IS 0.5
SIN OF 90 IS 1
>
```

SNDBRK

FORMAT: 1. SNDBRK "COMn:",<nombre>

Abréviation: SNDB.
Voir également:

BUT: Envoie un nombre spécifié de codes d'interruption au port série spécifié pour arrêter la transmission d'entrée depuis un autre ordinateur.

REMARQUES: Les ports sont spécifiés comme suit:

COM1: Port série RS-232C
COM2: Port série optique
COM: Port spécifié par la dernière commande SETDEV

Le paramètre <nombre> spécifie le nombre de caractères d'interruption à envoyer sous forme d'une chaîne continue au port dans une gamme de 1 à 255.

EXEMPLE:

```
>SNDBRK "COM1:",20
```

Enverra le caractère d'interruption 20 fois au port RS-232C afin d'arrêter la transmission de données. Il est souvent conseillé d'envoyer plus d'un caractère d'interruption, car l'autre ordinateur peut ne pas recevoir avec succès le premier caractère d'interruption.

SNDSTAT

FORMAT: 1. SNDSTAT "COMn:",[<protocole>],[<temporisation>]

Abréviation: SN.
Voir également: RCVSTAT

BUT: Spécifie le protocole de connexion d'émission pour le port RS-232C et la valeur de temporisation pour les deux ports série.

REMARQUES: L'un quelconque des ports suivants peut être spécifié:

COM1: Port série RS-232C
COM2: Port série optique
COM: Port actuellement choisi par la dernière instruction SETDEV

Les états de signaux auxquels seront envoyées les données sont spécifiés par la valeur du nombre décimal (0 à 255) désigné pour le paramètre <protocole>. Le nombre est intérieurement converti en nombre binaire de 8 bits et chaque bit spécifie la condition pour un signal de commande pour la connexion à exécuter. Si toutes les conditions sont satisfaites, la connexion est complet et la transmission de données peut être effectuée. Les bits 7 et 8 sont toujours programmé à 0 lorsqu'ils ne sont pas utilisés.

Bit	Valeur	Condition de signal
1	0	DTR haut
(bit le moins significatif)	1	DTR bas
2	0	RTS haut
	1	RTS bas
3	0	CTS haut
	1	CTS bas
4	0	CD haut
	1	CD bas
5	0	DSR haut
	1	DSR bas
6	0	CI haut
	1	CI bas

7Non utilisé; programmé à 0

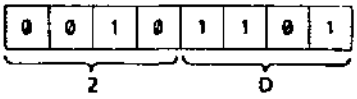
8Non utilisé; programmé à 0
(bit le plus significatif)

La valeur par défaut (valeur implicite) pour <protocole> est 59 (00111011), qui est le signal CTS est haut.
La temporisation correspond à la durée que l'ORDINATEUR attendra avant de décider que l'unité en attente n'est pas prête ou n'est pas disponible. Le paramètre optionnel <temporisation> spécifie la temporisation dans une gamme de 0 à 255, à des échelons de 0,5 seconde. Une temporisation de 0 invalide la temporisation de consigne (temporisation infinie) et est la valeur par défaut.

EXEMPLE:

```
>SNDSTAT "COM1:",45
```

Etablit les conditions de signal pour le port série RS-232C spécifiant que RTS et DSR doivent être hauts pour les données à envoyer. La valeur de temporisation est indéfinie. 45 en valeur décimale est 2D en hex et représente la configuration binaire suivante:



Notez que le bit le plus significatif ou le bit le plus à gauche est un bit 8, tandis que le bit le moins significatif ou le bit le plus à droite est un bit 1.

SORGN

FORMAT: 1. SORGN

Abréviation: SO.
Voir également: LUNE, GLCURSOR

BUT: Détermine la position actuelle du stylo en tant que point d'origine.

REMARQUES: En mode de graphique, SORGN spécifie la position actuelle du stylo comme point d'origine (0,0) pour toutes les commandes de graphique qui suivent. La position implicite (position par défaut) pour le point d'origine est la position d'origine gauche pour X et la position actuelle du papier spécifiée par les touches d'imprimante [M] et [M] pour Y.

EXEMPLE: Voir LUNE.

SQR

FORMAT: 1. SQR(X)

Abréviation: SQ. *Voir également:*

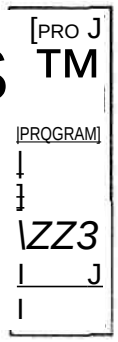
BUT: Renvoie la racine carrée d'expression X.

REMARQUES: Si l'expression a un nombre négatif, SQR(X) engendre un code d'erreur.

EXEMPLE:

```
>PRINT SQR(5)
             2.236067977
>
```

STATUS



FORMAT: 1. STATUS <valeur>

Abréviation: STA.
Voir également: MEM, NEW

BUT: Renvoie la place disponible et d'autres informations pour différentes zones de mémoire.

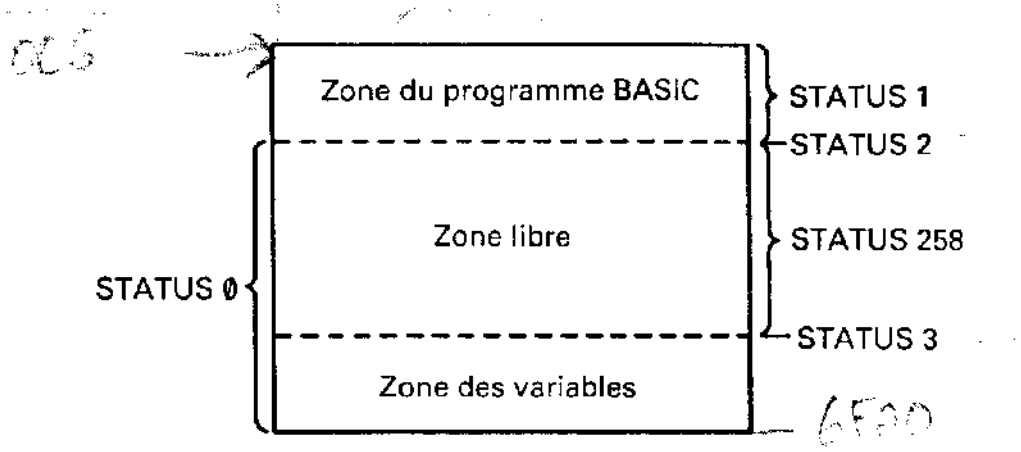
REMARQUES: Le paramètre <valeur> spécifie la zone de mémoire et les informations renvoyées comme suit:

<valeur>	Informations renvoyées
0	Mémoire disponible dans la zone d'utilisateur en octets (zone disponible + zone de variables).
1	Encombrement en octets d'un ou plusieurs programmes en cours de chargement
2	Adresse inférieure dans la mémoire de la zone d'utilisateur disponible.
3	Adresse supérieure dans la mémoire de la zone d'utilisateur disponible.
4 à 255	Dernier numéro de ligne exécuté pour le programme en cours d'exécution.
256	Numéro de bloc dans lequel existe l'adresse inférieure de la zone d'utilisateur disponible.
257	Numéro de bloc dans lequel existe l'adresse supérieure de la zone d'utilisateur disponible.
258	Mémoire non affectée dans la zone d'utilisateur en octets (zone disponible).
259	Mémoire non utilisée dans le module de programme dans le logement 1 en octets.
260	Mémoire non utilisée dans le module de programme dans le logement 2 en octets.

Topographie mémoire: RAM interne et logements d'extension

La topographie mémoire suivante représente l'affectation de la

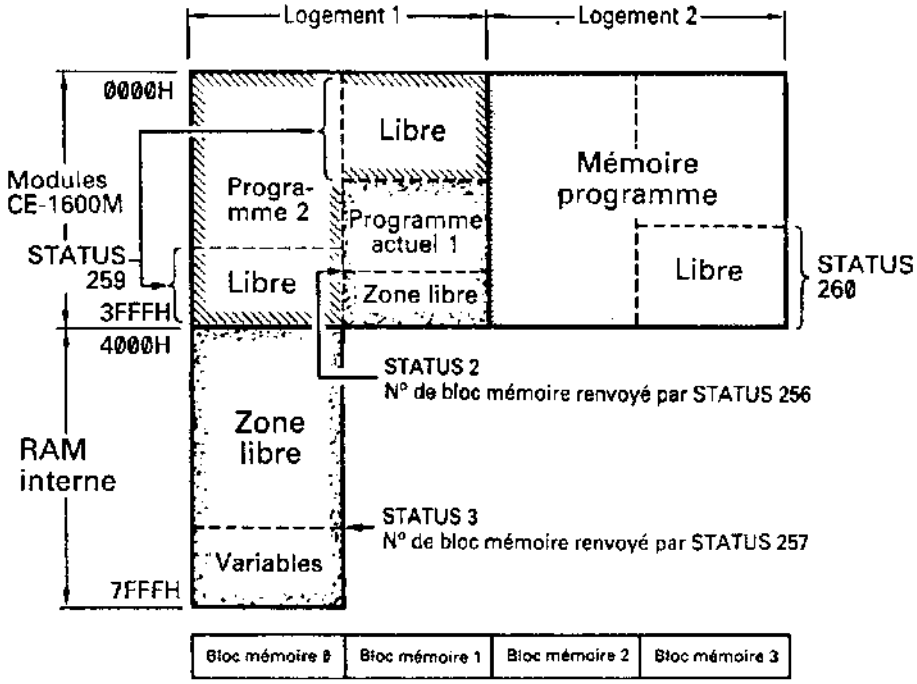
Topographie mémoire: Zone d'utilisateur RAM interne



mémoire avec:

(1)Un module CE-1600M dans le logement 1 avec la partie affectée seulement pour stockage de programme; la place restante est affectée pour extension de RAM interne (zone disponible).

(2)Un module CE-1600M dans le logement 2 avec la place affectée seulement pour stockage de programme.



EXEMPLE:

```
>STATUS 0
                              10789
```

La mémoire disponible est de 10789 octets.

STOP

FORMAT; 1. STOP

Abréviation: ST.
Voir également: CONT, END

BUT: Utilisée pour arrêter l'exécution, ordinairement durant la mise au point.

REMARQUES . L'instruction STOP est ordinairement introduite dans un programme à la phase de mise au point pour arrêter l'exécution, afin de permettre aux contenus de variables d'être examinés ou changés par entrée directe. Le message suivant est affiché:

BREAK IN <numéro de ligne>

L'exécution peut être recommencée par la commande CONT, à condition qu'aucune modification ne soit apportée au programme durant l'interruption. A la différence de l'instruction END, les fichiers ne sont pas fermés lorsqu'une instruction STOP est exécutée.

EXEMPLE:

```
10:FOR N=1 TO 50
20:LET S=N*5
25:STOP
30:LINE (0,0)-(N,S)
40:NEXT N

>RUN
BREAK IN 25
:
:
PRINT S
                              5
>CONT
:
```

[25J Met la ligne au point, compte tenu de la valeur de S.

FORMAT: 1. STR$ (<va!eur numérique>)

STR$

Abréviation: STR. *Voir également:* VAL

BUT; Convertit les données numériques en données de chaîne.

REMARQUES: La fonction STR$ change un nombre entier en une chaîne. La chaîne sera constituée par les mêmes chiffres en tant que nombre original, mais celui-ci est traité sous forme d'une chaîne de caractères SHARP ASCII dans un traitement subséquent. La fonction STR$ a un effet opposé à celui de la fonction VAL.

Si les données numériques sont négatives, la chaîne sera précédée par un signe "-". Si la <valeur numérique> est trop grande pour être maintenue dans la variable alphanumérique, elle est représentée dans une notation à virgule flottante.

EXEMPLE:

```
:
:
110:N=N*3
120:A$=STR$ (N)
130:B$=LEFT$ (A$,1)
140:M=VAL (B$)
:
:
```

[110] Le programme effectue certains calculs sur la variable numérique N.

[120] La variable numérique N est convertie en variable de chaîne A$. Les variables alphanumériques sont plus faciles à manipuler que les variables numériques. Dans cet exemple, imaginer que le premier chiffre du nombre est requis. C'est peut-être un code pour certains traitements subséquents. Une fois le nombre converti en chaîne, nous pouvons utiliser n'importe laquelle des commandes de manipulation de chaîne: INSTR, LEFT$, RIGHT$, MID$.

[130] Stocke seulement le premier chiffre du nombre, ou caractère car il est maintenant traité par le programme, dans la variable alphanumérique B$.

[140] Le monochiffre est reconverti en variable numérique de sorte qu'il puisse être traité par le programme en tant que nombre.

TAB

FORMAT: 1. TAB <colonne>

Abréviation:

Voir également: LCURSOR, LPRINT

BUT: En mode de texte, met le stylo d'imprimante sur la colonne spécifiée.

REMARQUES: La commande TAB est similaire à la commande LCURSOR; elle déplace le stylo à la colonne d'impression spécifiée dans la gamme allant de 0 à la longueur de ligne maximum spécifiée par l'instruction PCONSOLE en cours. L'emplacement de la colonne d'impression change sujvant la taille de caractère spécifiée par la commande CSIZE. TAB peut être également utilisée dans une instruction LPRINT.

Ce programme sera passé seulement en mode TEXT.

EXEMPLE:

```
5:TEXT
10:TAB 40
20:LPRINT "MILIEU"
30:LPRINT "GAUCHE"
40:END
```

(10] Met le chariot du stylo au milieu du papier.

(20] Ce message lancera l'impression à partir de la position actuelle de l'imprimante au milieu de la page.

[30] Après la dernière LPRINT, la tête d'imprimante revient à la gauche. Ainsi, cette ligne est imprimée dans la position normale gauche justifiée.

TAN

FORMAT: 1. TAN(X)

Abréviation: TA.

Voir également: ATN, COS, SIN

BUT: Renvoie la tangente de l'angle X

REMARQUES Cette fonction renvoie la tangente de l'angle X, où X est exprimé en degrés, radians ou gradients. Selon cette expression, le mode de l'ordinateur est spécifié par la commande DEGREE, RADIAN ou GRAD. Un code d'erreur est engendré si X est 90°.

EXEMPLE:

```
10:DEGREE
20:PRINT "TAN DE 0 EST";TAN (0)
30:PRINT "TAN DE 45 EST";TAN (45)

>RUN
TAN DE 0 EST 0
TAN DE 45 EST 1
>
```

TEST

FORMAT: 1. TEST

Abréviation: TE.
Voir également:

BUT: Effectue un essai sur l'imprimante dans toutes les couleurs.

REMARQUES: TEST émet des commandes à l'imprimante pour dessiner quatre cases avec quatre couleurs disponibles dans l'ordre de numéro de couleur (0 à 3): noir → bleu → vert → rouge. L'ordinateur est alors mis en mode de texte. Pour interrompre l'essai, appuyez sur la touche **BREAK**.

EXEMPLE:

```
> T ES T

Commence l'essai de l'imprimante pour dessiner les 4 cases.
```

FORMAT: 1. TEXT

Abréviation: TEX.
Voir également: GRAPH

TEXT

BUT: Met l'imprimante en mode de texte.

REMARQUES: La taille de caractère est spécifiée à 2 en mode de texte, et le point d'origine est réglé à ia position gauche du stylo.

L'exécution de la commande TEXT remet la commande PAPER et les paramètres <limite en sens inverse> et <limite en sens direct> à leurs valeurs par défaut.

EXEMPLE:

```
>TEXT
```

L'ordinateur choisît le mode TEXT pour l'imprimante.

TIME

FORMAT: 1. TIME
2. TIME = MMDDHH.mmss

Abréviation:
Voir également: TIMES

BUT: Spécifie et renvoie la date et l'heure pour l'horloge intégrée à l'ordinateur.

REMARQUES : La commande TIME renvoie l'heure actuelle dans le format MMDDHH.mmss où MM est un nombre de deux chiffres entre 01 et 12 représentant le mois, DD un nombre de deux chiffres entre 01 et 31 représentant le jour, HH un nombre de deux chiffres entre 00 et 23 représentant l'heure, mm un nombre de deux chiffres entre 00 et 59 représentant les minutes et ss un nombre de deux chiffres entre 00 et 59 représentant les secondes. L'horloge n'est pas affectée par la coupure d'alimentation. Elle n'indique pas l'année.

TiME=MMDDHH.mmss spécifie l'heure et la date actuelles sur l'horloge intégrée dans le même format que celui précité.

EXEMPLE:

```
>TIME=031220.3215
```

Spécifie l'heure et la date actuelles à 8:32 et 15 secondes de l'après-midi du 12 mars.

TIME$

FORMAT: 1. TIME$ = "HH:MM:SS" 2.
TIME$

Abréviation: TI.
Voir également: DATES, TIME$ ON/OFF/STOP, ON TIMES GOSUB

BUT: TIME$ est une variable de système qui contient l'heure de l'horloge incorporée dans l'ordinateur.

REMARQUES Dans le premier format, en tant qu'instruction, TIME$ spécifie la valeur de l'horloge temps réel. HH est un nombre de deux chiffres entre 00 et 23 représentant l'heure. MM est un nombre de deux chiffres entre 00, et 59 représentant les minutes. SS est un nombre de deux chiffres entre 00 et 59 représentant les secondes. Les deux-points sont utilisés pour séparer les trois introductions.

Dans le second format, TIME$ renvoie l'heure de l'horloge temps réel dans un format de HH:MM:SS au programme sous forme d'une variable de chaîne.

EXEMPLE:

```
>TIME$="09:15:00"
:
:
>PRINT TIME$
09:15:06
```

Spécifie l'heure et puis l'imprime.

TIME$ ON/OFF/STOP

FORMAT: 1. TIMES ON
2.TIME$ OFF
3.TIMES STOP

Abréviation: TI.
Voir également: TIMES, ON TIME$ GOSUB

BUT; Invalide ou valide à nouveau l'instruction ON TIME$ GOSUB.

REMARQUES: TIME$ OFF invalide toutes les instructions ON TIMES GOSUB de telle façon que les appels de sous-programme ne puissent jamais être effectués.

TIMES ON valide à nouveau toutes les instructions ON TIME$ GOSUB.

TIMES STOP invalide également toutes les instructions ON TIMES GOSUB mais lorsque TIMES ON est subséquemment exécutée. Si l'heure spécifiée dans TIMES = d'une instruction ON TIMES GOSUB est atteinte pendant que l'instruction TIMES STOP est valide, un renvoi sera immédiatement fait sur le sous-programme. La commande par défaut est TIMES STOP.

EXEMPLE : Voir ON TIMES GOSUB

TITLE

FORMAT: 1. TITLE "S0:"
2.TITLE "S1:"
3.TITLE "S2:"

Abréviation: TIT.
Voir éaalement:

BUT: Choisit le module de programme dans le logement 1 ou le logemet 2.

REMARQUES: Deux modules au maximum peuvent être montés dans les fentes sur le dos de l'ordinateur. Cette commande détermine quel logement est actuellement choisi.

TITLE :"S0:" choisit RAM interne de l'ordinateur. Il s'agit de la commande par défaut après une ALL RESET.
TITLE "S1:" choisit le module dans le logement 1. TITLE "S2:" choisit le module dans le logement 2.

Si le logement de module contient un module de disque RAM au lieu d'un module de programme, un message de code d'erreur est affiché.

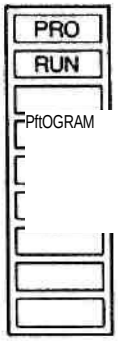
NOTE:
Dans certaines situations après exécution de TITLE durant l'arrêt de l'exécution d'un programme principal de mémoire, les touches [redacted] et [redacted] ne listeront pas le programme dans le module spécifié. Dans ce cas, utiliser LIST ou terminer d'abord l'exécution du programme principal de mémoire.

EXEMPLE:

```
>TITLE "SOI"
>
```

Choisit la mémoire principale de l'ordinateur pour usage après utilisation d'un module de mémoire dans l'un des logements.

TRON/TROFF



Abréviation: TR./TROF.
Voir également: CONT, STOP

FORMAT: 1. TRON
2. TROFF

BUT: Utilisée pour faire effectuer l'analyse par programme de mise au point ou pour l'annuler.

REMARQUES: L'analyse est effectuée avec des programmes de mise au point. Lorsque l'analyse est lancée, l'ordinateur s'arrête pendant 0,5 seconde après l'exécution d'une ligne de programme et l'affichage du numéro de ligne sur le côté droit de l'écran.

Si la touche ↓ est enfoncée pendant que le numéro de ligne est affiché, l'ordinateur passe à l'exécution en pas à pas. Et la touche ↓ doit être enfoncée chaque fois pour exécuter la ligne suivante. Si la touche ↓ est maintenue enfoncée, l'exécution est continuée de ligne en ligne sans affichage du numéro de ligne. L'instruction venant d'être exécutée peut être explorée sur toute sa longueur en appuyant sur la touche ▶ ou ◀. L'enfoncement de la touche ↑ rétablit une exécution continue (attente de 0,5 s ec. entre lignes).

Lorsque l'exécution est arrêtée à une instruction PRINT ou INPUT, l'enfoncement de la touche ENTER reprendra l'exécution.

Si l'exécution est interrompue par une instruction STOP ou par enfoncement de la touche BREAK, l'exécution de l'analyse continue peut être recommencée par enfoncement de la touche CA, ou l'exécution de l'analyse en pas à pas par enfoncement de la touche ↓.

L'analyse peut être également lancée ou arrêtée par un programme. Après TRON, l'analyse reste pour toute l'exécution subséquente jusqu'à ce que TROFF soit exécutée.

Le déroulement de ce programme après lancement de l'analyse affichera les numéros de ligne sur l'écran lorsque les lignes sont exécutées. TROFF arrête l'analyse, de sorte que les numéros de ligne ne sont plus

EXEMPLE:

```
10:FOR I=1 TO 3
20:PRINT "I=";I
30:NEXT I
40:END

>TRON
>RUN
                                10
I= 1                            20
                                30
I= 2                            20
                                30
I= 3                            20
                                30
>TROFF
>
```

indiqués dans un déroulement subséquent de programmes.

VAL

FORMAT: 1. VAL(X$)
2. VAL("<chaîne>")

Abréviation: V.
Voir également: HEX$, STR$

BUT: Convertit une chaîne de caractères numériques en valeur déci maie.

REMARQUES: La fonction de VAL est l'inverse de celles de STR$. Elle change une chaîne composée de caractères en valeur numérique qui peut être utilisée en tant que nombre dans un traitement subséquent.

Si la chaîne est décimale, elle doit être composée de caractères 0-9, avec point décimal optique et signe. Dans cette forme, la fonction de VAL est opposée à celle de STR$. Si la conversion n'est pas possbile du fait de l'inclusion de caractères interdits, la valeur 0 est renvoyée.

[10] Introduit une chaîne à convertir en valeur numérique.

EXEMPLE:

```
10:INPUT "FREQUENCE DE CYCLE ":A$
15:IF ASC (A$)<480R ASC (A$)>57 THEN 100
20:F=VAL (A$)
30:PRINT F
40:STOP
:
:
100:PRINT "DOIT ETRE UN NOMBRE":GOTO 10
```

[20] La chaîne est convertie en valeur numérique équivalente. L'introduction d'un nombre sous forme d'une chaîne donne l'occasion au programmateur de vérifier que la chaîne introduite est du type correct ou est comprise dans la gamme correcte dans le programme lui-même.

WAIT

FORMAT: 1. WAIT <durée> [,P]
2.WAIT <durée> [,S]
3.WAIT

Abréviation: W.
Voir également: PRINT, GPRINT, LINE

BUT: Spécifie la durée pendant laquelle l'exécution est arrêtée après exécution d'une instruction PRINT, GPRINT ou LINE.

REMARQUES: WAIT est utilisée pour afficher les données sur l'écran pendant une durée fixée avant de continuer l'exécution.

WAIT <durée> arrête l'exécution après une instruction PRINT, GPRINT ou LINE pendant une durée spécifiée dans une gamme de 0 à 65535, à des échelons d'environ 1/64^{e} de seconde. WAIT 64 arrête l'exécution et il en résulte que l'instruction est affichée sur l'écran pendant environ 1 seconde.

Les options P et S peuvent être spécifiées seulement en MODE 0 (mode PC-1600).

L'option P met l'exécution en attente après l'instruction PRINT, GPRINT ou LINE pendant la durée spécifiée. L'option S spécifie que la durée de WAIT est appliquée au roulement des informations sur l'écran: une fois l'écran complet des données (4 lignes) imprimé, l'écran ne commencera pas à faire rouler l'image jusqu'à ce que la durée spécifiée soit terminée. La valeur par défaut est P.

La spécification de WAIT seule détermine la durée d'attente à l'infini; l'ordinateur arrêtera l'exécution jusqu'à ce que la touche soit pressée.
ENTER

Lorsque l'exécution est lancée par une commande RUN, les valeurs par défaut sont les suivantes:

0 en MODE 0 (mode PC-1600); aucun arrêt dans l'exécution suivant une instruction PRINT.

INFINITY en MODE 1 (mode PC-1500); exécution arrêtée jusqu'à ce que la touche **ENTER** soit pressée.

EXEMPLE:

```
20:WAIT 200
30:PRINT "LIRE CECI RAPIDEMENT"
40:WAIT
50:PRINT "MAINTENANT C'EST FINI"
60:END

>MODE 1
>RUN
```

[20] Spécifie la temporisation pendant laquelle l'exécution s'arrête (environ 3 secondes) après chaque PRINT.
[30] Imprime en trois secondes le message qui est superposé à la ligne 50.
[40] Réinitialise les conditions WAIT aux valeurs par défaut.
[50] Le second message se superpose au premier et reste sur l'écran (valeur par défaut de WAIT).

WAKE$

FORMAT: 1

	WAKE$ (0)	
3	WAKE$(1)	" <heure>:<chaîne de commande>
	WAKE$ (0)	
4	WAKE$(1)	" <chaîne de commande>"

n

Abréviation: WAK. *Voir également:* KBU F F$

n

n

BUT: Spécifie l'heure de réveil et la chaîne de commande pour la mise sous tension automatique.

REMARQUES: Dans le premier format, WAKE$ permet à l'ordinateur d'être programmé pour se mettre sous tension à l'heure spécifiée et exécuter une commande lors de la mise sous tension. Combinée avec la commande KBUFF$, la commande WAKE$ permet une exécution automatique de fichiers séquentiels à l'heure spécifiée. L'<heure> est spécifiée dans le format MM/DD/HH/mm.

Dans le second format, WAKES permet à l'ordinateur d'être programmé pour se mettre sous tension et d'exécuter une commande lorsque la ligne de signal Cl (broche 9) provenant de l'interface RS-232C devient "haute", et exécute la <chaîne de commande> spécifiée. Voir la partie III du chapitre 6 concernant l'interface RS-232C.

La ligne <cfiaîne de commande> doit être terminée par le code hexadécimal pout la touche d'introduction; c'est-à-dire, CHR$(&0D) comme dans l'exemple:

WAKES (0) - " 10/03/08/00; RU N" + CHRS (&0D)

La longueur maximum de la commande est de 26 caractères. Les formats 3 et 4 annulent la spécification des formats 1 et 2 respectivement.

EXEMPLE:

```
>WAKE$ (0) =" 12/25/07/30:RUN " +CHR$  (&0D)
10:PRINT "BONJOUR ET.... "
20:PRINT "... JOYEUX NOEL... "
:
:
```

WAKES mettra l'ordinateur sous tension à 7:30 la matin du jour de Noël et exécutera le programme actuellement en mémoire. Ici, le programme imprimera les bons roeux, etc.

XCALL

FORMAT: 1. XCALL <adresse> [,<nom de variable>]

Abréviation: XC.

Voir également: NEW, POKE, XPOKE

BUT: Utilisée pour appeler un programme de langage machine.

REMARQUES: L'instruction XCALL est essentiellement identique à l'instruction CALL et appelle un programme ou sous-programme de langage machine stocké en mémoire.

La différence est que cette commande est compatible avec la commande spécifiée danp le PC-1500. Lorsque le programme est appelé, une valeur de variable simple peut passer du programme BASIC en cours d'exécution en programme de langage machine. Et lorsque l'exécution du programme de langage machine est terminée, la même variable simple sera renvoyée au programme d'appel. Les registres A et X sont utilisés à cet effet, la variable à passer doit avoir été affectée avant l'exécution de XCALL. Le programme de langage machine doit avoir été introduit en mémoire par l'instruction XPOKE avant qu'il ne puisse être appelé.

Le paramètre <adresse> spécifie l'adresse inférieure dans le bloc de mémoire actuel où le programme de langage machine est stocké.

Le paramètre <nom de variable> spécifie la variable dont la valeur est passée au programme de langage machine à l'entrée, et renvoyée au programme d'appel BASIC à la sortie si le drapeau de report est élevé.

Le paramètre <nom de variable> spécifie une variable numérique, et doit être un nombre entier dans une gamme de -32768 à 32767. La valeur est amenée au registre X, et le contenu du registre X est ramené au programe d'appel avec le même nom de variable à la sortie. Si un nom de variable de deux caractères est utilisé, il doit avoir été déclaré par l'instruction DIM ou un code d'erreur sera engendré.

cernent de variable alphanumérique et le registre A maintient la longueur de chaîne.

EXEMPLE:

```
:
:
400:XCALL 57405, X
410:PRINT "LA VALEUR DEX RENVOYEE"
420:PRINT "DU LANGAGE MACHINE EST";X
:
:
```

[400] Appelle le programme de langage machine maintenu en mémoire en lançant l'exécution à l'adresse de mémoire 80.
[400-420] XCALL passe à travers la variable numérique "X" et la valeur renvoyée est imprimée ici.

XPEEK

FORMAT: 1. XPEEK <adresse> 2.
XPEEK #<adresse>

Abréviation: XP.
Voir également: PEEK, POKE, XPOKE

BUT: Renvoie un octet de données depuis l'adresse spécifiée dans la mémoire.

REMARQUES: La commande XPEEK est essentiellement identique à la commande PEEK, mais le format est compatible avec la commande désignée dans PC-1500.

Le paramètre <adresse> spécifie une adresse dans la zone de mémoire 0 (ME 0) à solliciter.

Le paramètre #<adresse> spécifie une adresse dans la zone de mémoire 1 (ME 1) à solliciter. Le bloc mémoire utilisé est le bloc actuellement sollicite.

Pour de plus amples détails, voir la topographie mémoire en annexe.

EXEMPLE:

```
>XPEEK100
>                    37
```

Renvoie les données à l'adresse de mémoire 100 (&64). La valeur actuelle est 37.

XPOKE

FORMAT: 1. XPOKE odresse de début>,<liste de nombres entiers> 2. XPOKE #<adresse de début>,<liste de nombres entiers>

XPO *Voir également:* PEEK, POKE, XPEEK

BUT: Introduit un ou plusieurs octets de données en mémoire en commençant par l'adresse spécifiée.

REMARQUES: La commande XPOKE est essentiellement identique à la commande POKE, mais le format est compatible avec la désignation de commande de PC-1500.

Le paramètre odresse de début> spécifie l'adresse où le premier octet sera introduit dans la zone de mémoire 0 (ME 0). Les octets le suivant seront introduits aux adresses subséquentes.
Le paramètre #<adresse de début> spécifie l'adresse dans la zone de mémoire 1 (ME 1).

Le paramètre <liste de nombres entiers> spécifie la liste de valeurs entières dans une gamme de 0 à 255 (&0 à &FF) à introduire aux adresses consécutives depuis l'adresse spécifiée. Les articles dans la liste sont séparés paç des virgules.

Le bloc de mémoire utilisé est le bloc actuellement sollicite.

Pour de plus amples détails, voir la topographie mémoire en annexe.

EXEMPLE:

```
>XPOKE100, 255, 255
>
```

En partant de l'adresse de mémoire 100 (&64), les deux valeurs 255 sont introduites en mémoire. 255 est &FF. Ceci aura comme résultat de fixer 16 bits de mémoire à "1". Ne pas pratiquer l'utilisation de cette commande sans étudier la topographie mémoire et sans savoir où il faut écrire. Cet exemple n'étant qu'une démonstration, soulèvera des problèmes au niveau de l'exécution. Toutefois, une 'ALL RESET permettra le rétablissement après la repoduction des erreurs dues à un mauvais relevé de la mémoire.

PARTIE V

ANNEXES

A. REMPLACEMENT DES PILES
B. REMPLACEMENT DES MODULES DE RAM
C. TABLEAU DES CODES DE CARACTERES
D. TOPOGRAPHIE MEMOIRE
E. PROGRAMMES LANGAGE MACHINE
F. CODES ERROR POUR LE PC-1600
G. LISTE DES COMMANDES BASIC
H. COMPATIBILITE AVEC LE PC-1500 ET SES PERIPHERIQUES
I. ENTRETIEN ET PANNES J. SPECIFICATIONS

Annexe A

Remplacement des Piles

Vous pouvez savoir de deux manières que les piles de votre PC-1600 sont pratiquement épuisées:

•Un témoin lumineux d'épuisement des piles, symbole BATT, s'allume sur la ligne d'état de l'écran du PC-1600. Le symbole BATT peut également apparaître si les piles de l'imprimante sont faibles, quand celle-ci est connectée à l'ordinateur. Reportez-vous à la section BATT.

•Durant l'exécution, l'ordinateur peut afficher un code ERROR sur l'écran pour indiquer l'épuisement des piles. N'oubliez pas que les codes ERROR sont également affichés si les pUes de l'équipement connecté, comme l'imprimante par exemple, sont faibles. Voyez le résumé des codes ERROR dans l'Annexe F et les listes des codes ERROR dans les sections des équipements en option au Chapitre 6.

Notez aussi que la caractéristique Sauvegarde de mémoire, qui préserve toutes les données dans la mémoire de l'ordinateur au moment où vous mettez l'appareil hors tension, n'est pas valable lorsque les piles sont mortes ou remplacées. Ceci concerne aussi bien les données des modules mémoires lors de leur montage que celles de la mémoire interne de l'ordinateur. Les modules de programmes sont protégés de cette perte de données à cause de leurs piles de sauvegarde incorporées.

Si vous n'avez actuellement aucune donnée que vous ne désiriez conserver en mémoire, reportez-vous à la section sur la mise en place des piles. Remplacez les piles en respectant les instructions, puis voyez la section sur la réinitialisation de l'ordinateur et suivez la procédure ALL RESET pour réinitialiser l'ordinateur et nettoyer la mémoire.

Si vous avez des données en mémoire que vous désirez conserver, vous disposez de deux manières pour éviter la perte des données lors du remplacement des piles:

- Connectez un adaptateur CA sur l'ordinateur pour maintenir l'alimentation tandis que vous remplacez les piles. Procédez de la manière suivante:

1Mettez l'ordinateur hors tension
2Branchez l'adaptateur CA tout d'abord dans la prise murale, puis sur l'ordinateur
3Remplacez les piles
4Débranchez l'adaptateur CA
5Remettez l'ordinateur sous tension et reprenez lès opérations. Les données et programmes sont intacts. Il n'est pas nécessaire que vous réinitialisiez l'ordinateur.

La seconde procédure est un peu plus compliquée.

- Sauvegardez les données et programme sur un disque, une cassette ou un disque RAM avant de remplacer les piles. Après la mise en place des nouvelles piles, rechargez données ou programmes dans la mémoire pour continuer à travailler. Suivez les phases mentionnées ci-dessous:

  1Si aucun équipement externe n'est connecté, mettez l'ordinateur hors tension.
  2Connectez l'unité de disque, l'enregistreur de données ou le module programme (disque RAM) optionnels.
  3Mettez l'ordinateur sous tension et utilisez la commande SAVE (voir le Dictionnaire des commandes BASIC) pour écrire les données en mémoire dans l'un des équipements mentionnés.
  4Mettez hors tension et remplacez les piles.
  5Reportez-vous à la section sur la Réinitialisation de l'ordinateur et suivez la procédure pour réinitialiser l'ordinateur.
  6Utilisez les commandes LOAD pour recharger les données depuis l'équipement externe dans la mémoire de l'ordinateur, afin de continuer les opérations.

*

Annexe B

Remplacement des Modules de RAM

Il y a deux types de modules de RAM: les modules de programmes avec leur pile de sauvegarde incorporée pour préserver le contenu des mémoires lorsqu'ils sont enlevés et les modules de mémoires qui ne sont pas sauvegardés et sont utilisés pour agrandir la zone utilisateur de la mémoire interne de l'ordinateur.

Les modules de programmes ne nécessitent pas de précautions spéciales lors de leur remise en place et peuvent être enlevés ou installés à volonté.

Quand les modules de mémoire sont enlevés de l'ordinateur, les données sont alors perdues.

Après avoir mis en place un module de mémoire et mis l'ordinateur sous tension, le message N E W © ? : C H E C K s'affiche sur l'écran pour vous dire que l'ordinateur reconnaît un changement dans la capacité de la mémoire, par l'addition de ce module. Pour nettoyer la mémoire entière, du module de RAM comme de la RAM interne de l'ordinateur, pressez la touche H et assurez-vous que le mode spécifié est PROgramme. Sinon, pressez la touche Ml- Puis tapez N E W 0 et pressez la touche | |. Ce message n'est pas affiché et le vidage n'est pas nécessaire lorsque le module mis en place est un module de programme. Notez que la commande N E W vide le contenu de la mémoire mais ne vide pas les chaînes des touches de fonctions du mode RESERVE si exécutées en mode PROgramme. Par conséquent, après avoir mis en place un module, la zone RESERVE peut contenir des chaînes de touches de fonctions non nettoyées. Remettez-les à zéro en exécutant la commande NEW en mode RESERVE.

A moins qu'ils ne soient spécifiés avec la commande TITLE, les modules de programmes sont indépendants de la RAM interne de l'utilisateur lors de leur mise en place.

Vous pouvez utiliser l'instruction MEM en BASIC pour afficher la capacité disponible dans la mémoire libre de l'utilisateur après avoir mis en place un module de RAM, ou en n'importe quelle circonstance. Tapez MEM puis pressez la touche ENTER. Un chiffre s'affiche alors à l'extrême droite de la ligne suivante,
indiquant le nombre d'octets disponibles dans la mémoire libre de l'utilisateur. Par exemple, quand il n'y a pas de module dans les logements, l'entrée de M E M donne en réponse 11834, le nombre d'octets disponibles dans la RAM interne de l'ordinateur sans données.

Quand un module d'extension de mémoire est installé, un texte BASIC est écrit dans les zones de mémoire d'extension comme suit: S2: -* S1: -» RAM interne

Annexe C

Tableau des Codes de Caractères

Tableau des codes de caractères en mode 0

En mode 0, le jeu des caractères comprend les graphiques, les symboles grecs et les caractères internationaux en plus des symboles et caractères habituels en majuscules et minuscules. Ce jeu de caractères est similaire au jeu de caractères de l'ordinateur individuel IBM.

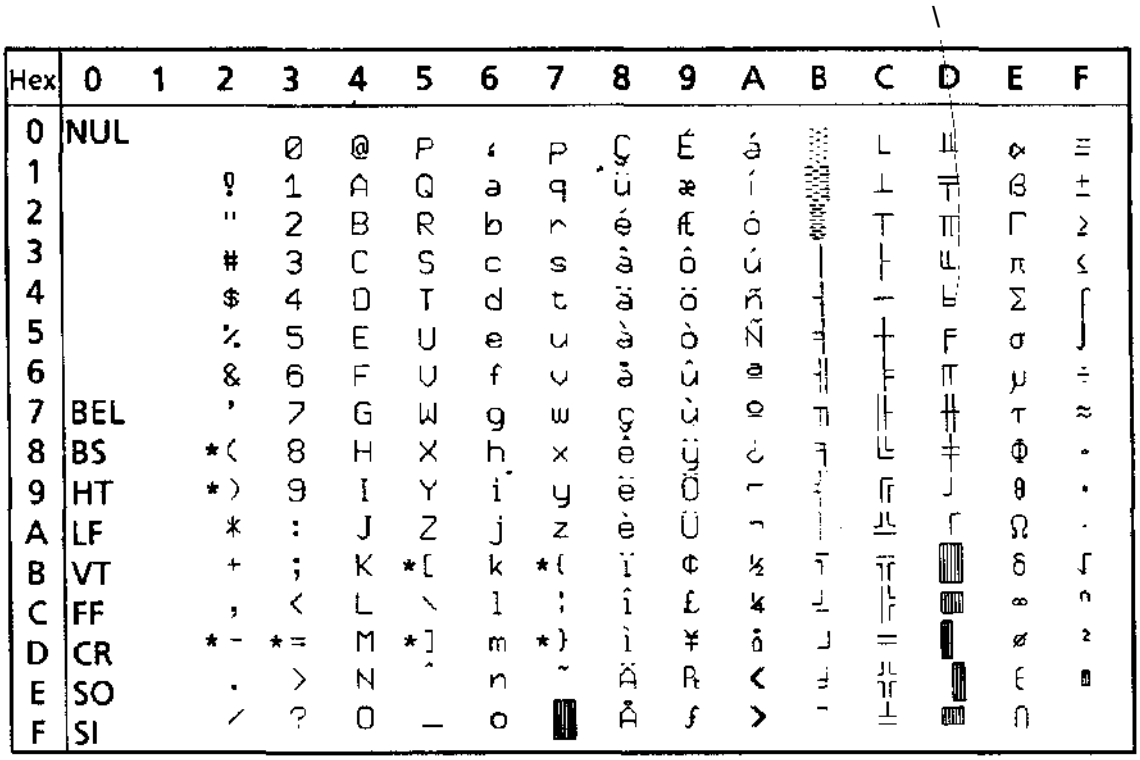


Tableau des codes de caractères du PC-1600

*Ces caractères ont une rotation sur 90° durant l'impression verticale spécifiée avec la commande ROTATE.

Pour trouver le code hexadécimal correspondant à un caractère dans le tableau, écrivez le nombre en tête de la colonne dans laquelle le caractère apparaît, suivi du nombre à gauche de la rangée dans lequel ce caractère est. Le code hexadécimal pour "a" est par conséquent &61.

Tout caractère dans ce tableau peut être imprimé sur l'écran à l'aide de la fonction CHR$ (<code de caractère>) en BASIC. Pour de plus amples détails, voyez CHR$ dans le Dictionnaire des commandes.

Jeu des caractères internationaux

Dans le tableau des codes de caractères, des caractères internationaux, de la valeur de code &80 à la valeur de code &A8, sont assignés aux touches alphanumériques du clavier en pressant la touche KBII, à droite des six touches de fonctions. Un guide des caractères internationaux figure en dessous des touches correspondantes.

Tableau des codes de caractères en mode

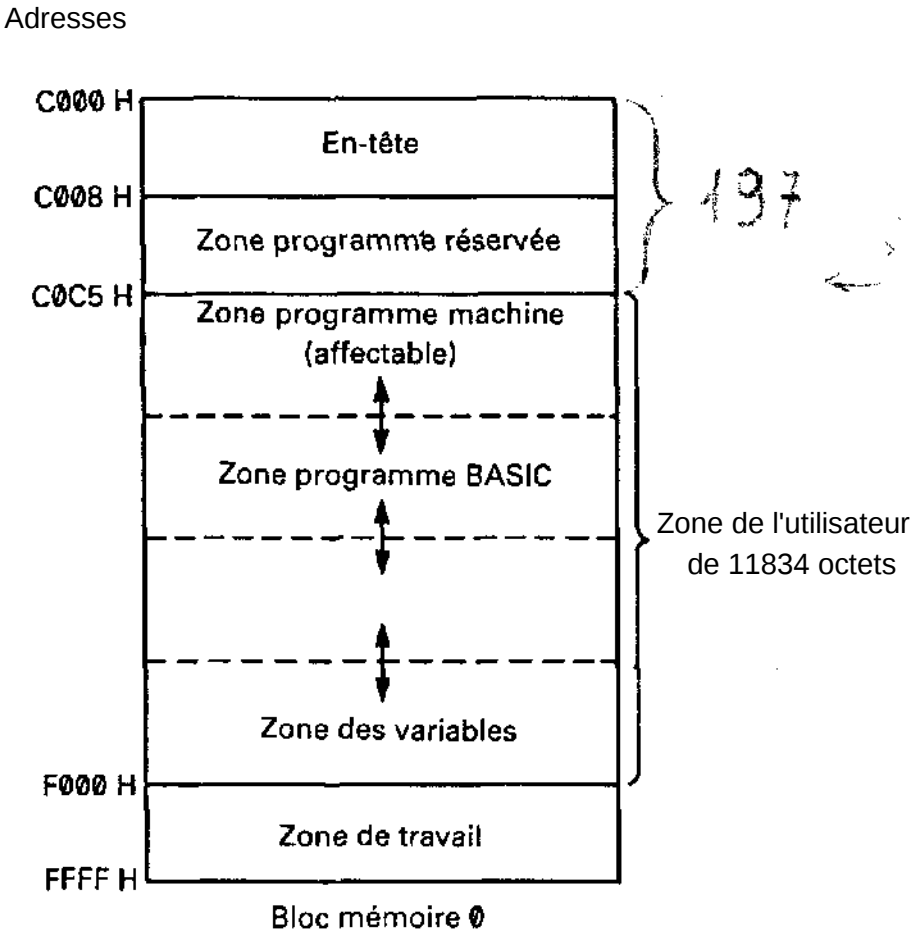
En mode 1, le jeu de caractères du PC-1600 est modifié pour être compatible avec le jeu plus limité des caractères d*u* PC-1500. Le tableau ci-dessous donne la valeur hexadécimale de position des caractères et le caractère correspondant pour le mode 1.

HEX.	27	5B	5C	5D	5E	5F	60	7B	7C	7D	7E	7F
MODE 0	°	[	\	]	^	_	`	{	|	}	~	
MODE 1		√	¥	π	^						~	

Annexe D

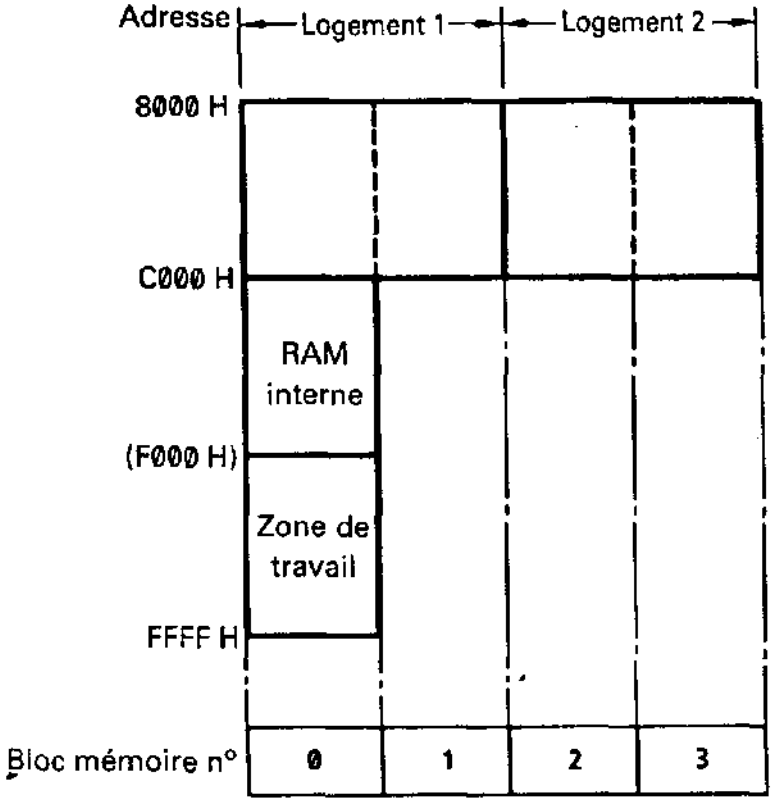
Topographie Mémoire

Le PC-1600 peut adresser 8 banques de mémoire (banques de 0 à 7). Les quatres premières banques sont allouées à la RAM. Les banques de 4 à 6 sont occupées par les ROM de système et la mémoire périphérique. La banque 7 est inoccupée mais adressable.

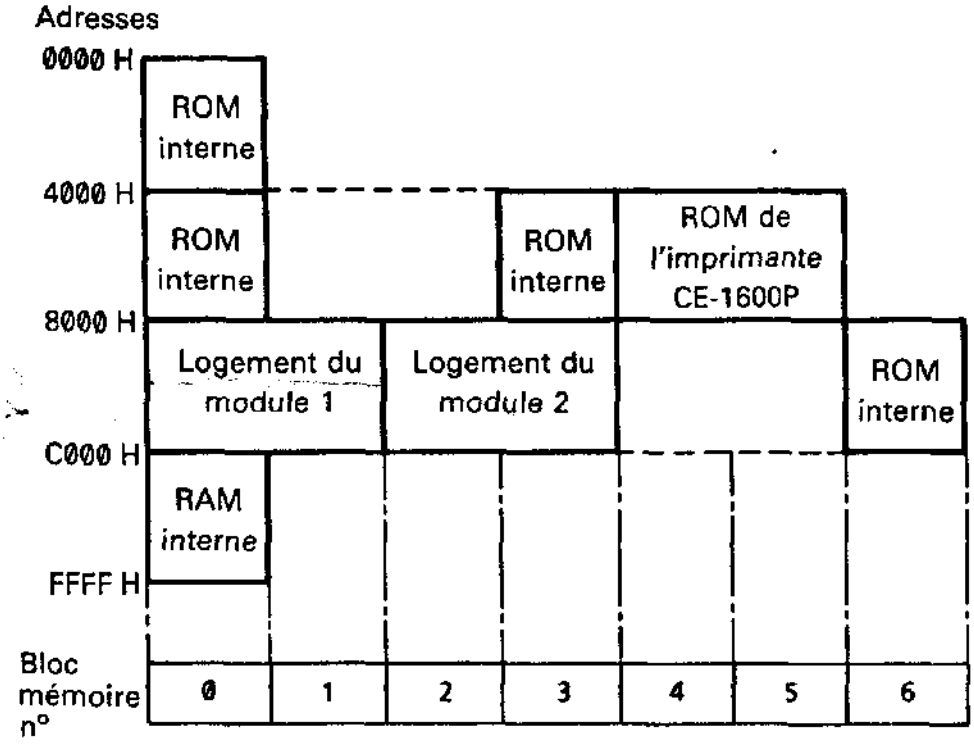


RAM interne [vu du processeur Z-80A]

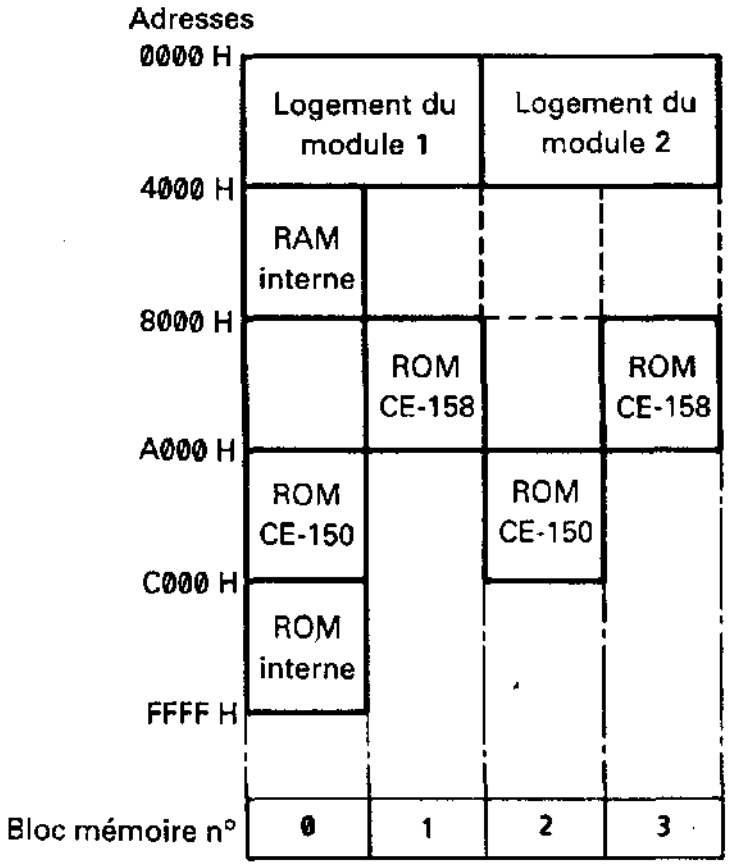
Sans module dans les logements d'extension, la RAM interne dispose de 11834 octets pour la zone de l'utilisateur. La topographie mémoire ci-dessus montre une partie de la zone de l'utilisateur allouée aux programmes langage machine. Cette zone de langage machine peut être spécifiée à 0 par la commande NEW.



Logements de RAM interne et d'extensions (blocs mémoire 0 à 3) [vu du processeur Z-80A]



Topographie mémoire générale [vu du processeur principal Z-80A]



Topographie mémoire générale [vu du processeur secondaire LH-5803]

/

Annexe E

Programmes Langage machine

\.

Le PC-1600 peut être programmé directement dans un code assembleur Z80A par un programmeur avancé

Commandes se rapportant au langage machine BASIC

Le BASIC supporte un certain nombre de commandes pour charger, sauvegarder, accéder et appeler les programmes langagemachine, ou pour commander les ports E/S machine directement. Certaines désignent le processeur principal Z80A et certaines, le processeur secondaire Ltf-5803. Ce sont:

BLOAD, BSAVE, CALL, CLOADM, CSAVEM, INP, OUT, PEEK, POKE, XCALL, XPEEK, XPEEK#, XPOKE, XPOKE#

Voyez le Dictionnaire des commandes pour les descriptions détaillées.

Affectation mémoire des programmes langage machine

La RAM interne peut être affectée aux programmes langage machine avec la commande NEW, qui positionne l'adresse la plus basse de la zone programme du BASIC. Vous pouvez également accéder aux programmes utilitaires du système dans d'autres zones de mémoires ou sur les ROM des équipements périphériques, mais ceci demande une connaissance détaillée de l'allocation mémoire du PC-1600, supérieure à celle couverte par ce manuel. Référez-vous aux Topographies mémoire de l'Annexe D.

Compatibilité avec le PC-1500

Le tableau suivant liste les commandes se rapportant au langage machine du PC-1600 qui sont différentes celles du PC-1500:

COMMANDE		FONCTION
PC-1600	PC-1500	
XCALL	CALL	Exécute le programme langage machine pour le processeur secondaire LH-5801/3
CALL		Exécute le programme langage machine dans le processeur principal (Z80A) du PC-1600
XPOKE	POKE	Ecrit les données dans l'espace mémoire du LH-5801/3
POKE		Ecrit les données dans l'espace mémoire du processeur principal Z80A
XPEEK	PEEK	Lit les données depuis la zone mémoire du LH-5801/3
PEEK		Lit les données depuis la zone mémoire du processeur principal Z80A
XPOKE#	POKE#	Envoie les octets de données au port E/S machine du LH-5801/3
XPEEK#	PEEK#	Renvoie les octets de données du port E/S machine du LH-5801/3

Annexe F

Codes Error pour le PC-1600

Codes des erreurs	Description

ERREURS AVEC LES PROGRAMMES

1 Erreur de syntaxe. L'instruction n'est pas conforme aux règles du BASIC Sharp.
Commande de programme utilisée en mode direct ou vice-versa lors de l'utilisation des périphériques du PC-1500 (CE-150, CE-158, CE-162E) en mode 1. Numéro de ligne hors gamme.
Le numéro de ligne dans le programme dépasse-65279.

2 NEXT sans FOR. Une instruction NEXT est rencontrée sans l'instruction FOR correspondante. Ceci peut provenir d'un mauvais emboîtement des boucles FOR/NEXT, la variable dans l'instruction NEXT n'étant pas identique à la variable spécifiée pour l'instruction FOR, ou, plus d'une instruction NEXT a été spécifiée pour une seule instruction FOR.

4READ sans DATA. Une instruction READ est rencontrée mais sans instruction DATA dans le programme. Indisponibilité de donnée. Il n'y a plus aucune donnée à lire sur la ligne DATA.

5Variable en tableau déclarée deux fois. La variable en tableau déclarée maintenant l'a déjà été précédemment.

6Variable en tableau non déclarée. La variable en tableau recontrée dans le programme n'a pas été déclarée avec l'instruction DIM.

7Erreur d'assortiment. Le type de variable ne concorde pas avec le type de donnée. Une tentative a été faite pour assigner une valeur numérique à une variable alphanumérique ou vice-versa.

8 Plus de deux dimensions déclarées pour une variable en tableau.

9 Index hors limite. L'index spécifié dans l'instruction n'est pas dans la limite spécifiée par l'instruction DIM.

Codes des erreurs	Description
10	Zone mémoire de variables pleine. Impossible de stocker plus de variables.
11	Numéro de ligne non défini dans les instructions* GOTO, ON..GOSUB ou ON..GOTO. Le numéro de ligne spécifié pour le branchement n'existe pas.
12	Chaîne alphanumérique de format incorrect, spécifiée dans l'instruction USING.
13	Mémoire pleine. Mémoire utilisateur insuffisamment disponible pour la taille du programme ou le stockage des variables alphanumériques dans le programme.
14	Pile pleine. Les boucles FOR/NEXT sont emboîtées trop profondément dans le programme. Pile pleine durant le calcul d'une fonction.
15	Pile pleine. Les sous-programmes sont emboîtés trop profon dément dans le programme. Longueur de la chaîne supérieure à 80 octets.
16	Présence d'un exposant supérieur à 99.
17	Types de données illégales en instruction arithmétique. Essayez d'exécuter les opérations arithmétiques en variable alphanumérique.
18	Spécification incorrecte des paramètres dans une fonction.
19	Paramètre hors limite.
20	Spécification incorrecte du nom de la variable fixe.
21	Expression numérique utilisée au lieu d'une variable.
22	Mémoire pleine. Ne peut charger le programme.
23	Spécifications incorrectes en temps dans la chaîne TIME.
24	Tentative effectuée pour sauvegarder, lister ou éditer un programme sur lequel un mot de passe a été spécifié à l'aide de la commande PASS.

Codes des erreurs	Description
25	Adresse invalide spécifiée avec l'instruction NEW.
26	Mode invalide. Ne peut exécuter la commande dans le mode actuel.
27	Commande illégale. Le périphérique adressé par cette commande n'est pas connecté.
28	Une instruction INPUT ou AREAD a rencontré une donnée de chaîne illégale qui recopie le nom d'une commande ou le nom d'une fonction.
30	Numéro de ligne hors limite. Le numéro de ligne dans le programme excède 65539 quanol le PC-1600 est utilisé en mode 1 avec les périphériques du PC-1500.
32	La position du curseur spécifiée par GCURSOR empêche l'affichage de la donnée introduite en réponse à l'instruction actuelle INPUT.
35	Le périphérique PC-1500 n'est pas connecté.
36	Incapable d'afficher la donnée comme spécifié dans le format d'édition à l'aide de l'instruction USING.
37	Dépassement de capacité. Le résultat du calcul excède 9,999999999E99.
38	Division par zéro. Zéro est utilisé comme diviseur parce que la variable ou l'expression était zéro à cet endroit du programme. Variable non définie utilisée comme diviseur.
39	Appel de fonction illégal. Un nombre négatif est spécifié en tant que nombre de la fonction SQR ou LN; Un nombre décimal est spécifié au lieu d'un nombre entier dans cer taines fonctions.
100	Renumérotation impossible avec la commande RENUM.
101	Nom d'unité invalide spécifié avec les instructions TITLE ou NEW.
102	Sélection d'uniré invalide (équipement non connecté).

Codes des erreurs Description

103 Mémoire pleine pour le module de RAM. Les paramètres INIT ne peuvent être spécifiés.

104 Plus de huit interruptions de programme spécifiées*dans le programme (instructions ON GOSUB).

105 Trop de niveaux d'interruptions de programme.

106 L'instruction RETI est trouvée sans l'instruction correspondante ON..GOSUB..

107 L'instruction RESUME est trouvée sans l'instruction correspondante ON ERROR GOSUB.

108 Le mot de passe ne peut être supprimé.

109 Nom de variable illégal dans l'instruction ERASE.

110 Ne peut spécifier le MODE 1 (mode PC-1500). Commande invalide en MODE 0. Les commandes associées avec les périphériques du PC-1500 ne fonctionnent qu'en MODE 1.

111 Mode invalide sur le PC-1600.

112 Ligne trop longue. Durant l'édition du programme, la longueur de la ligne logique excède 80 caractères.

131 Donnée de chaîne précédée du signe + ou -.

ERREURS AVEC LES CASSETTES

40 Erreur de syntaxe dans la commande de bande.

42 Mémoire disponible insuffisante. Pas assez d'espace mémoire pour charger le programme.

43 Erreur vérification de bande. Après l'exécution de CLOAD?, la copie en mémoire n'est pas conforme avec l'original sur la bande. Erreur de lecture de bande possible. Chargez et vérifiez à nouveau. Erreur d'assortiment de type avec l'instruction INPUT#. Le type de variable et le type de donnée ne correspondent pas. Tentez de lire les données en chaîne dans une variable numérique ou vice-versa.

Codes des erreurs	Description
44	Erreur de bande. Enregistreur incompatible; erreur de lecture de bande; ajustez les réglages de l'enregistreur (tonalité et volume).

ERREURS AVEC L'IMPRIMANTE

70 Stylo hors des limites -2048 à 2047 pour X ou Y.

71 Alimentation en papier inversée sur plus de 10,24 cm en mode TEXT. (Seulement pour l'interface à cassette et l'imprimante CE-150).

72 Spécification incorrecte du paramètre TAB.

73 Commande illégale en mode actuel. Commande graphiques utilisée en mode TEXT ou vice-versa.

74	Paramètres trop nombreux dans l'instruction LLINE ou RL1NE.
76	La sortie de ligne sur l'imprimante dépasse la longueur de ligne spécifiée avec PCONSOLE, ou le résultat d'un calcul numérique est trop long pour tenir sur une ligne en mode TEXT.

78 Ne peut exécuter les commandes LPRINT ou LLINE. Le stylo n'est pas en position ou l'imprimante est bloquée à cause d'une tension trop basse des piles.

79 Les signaux couleurs ne sont pas transmis à l'imprimante. (Seulement pour l'interface à cassette et l'imprimante CE-150).

80 Piles épuisées. L'imprimante est bloquée à cause d'une tension trop basse des piles.

PORTS SERIE E/S [COM1: et COM2:]

140 Paramètres invalides spécifiés dans l'instruction SETCOM.

141 La capacité du tampon de réception spécifié dans l'instruction INIT est trop large (supérieure à 16383 octets ou supérieure à la mémoire disponible).

Codes des erreurs Description

142 Erreur dans la réception de donnée par le port série (erreur de parité, erreur de dépassement, erreur de synchronisation, erreur intégrale de tampon de réception).

143 Erreur de délai d'attente. Aucune réponse d'une autre partie pendant une durée plus longue que la valeur de (temporisation) spécifiée dans les instructions RCVSTAT.

144 Le port série spécifié dans l'instruction SETDEV est déjà ouvert.

ERREURS AVEC LES FICHIERS

150 Trop de fichiers spécifiés dans l'instruction MAXFILES.

151 Le fichier existe déjà. Utilisez un autre nom de fichier.

152 Le fichier n'est pas trouvé. Vérifiez que le nom du fichier spécifié est correct.

Erreur de disque plein lors de la sauvegarde sur le disque avec la commande SAVE.

153 Spécification du numéro de fichier incorrecte dans l'instruction écriture ou lecture de fichier. Le fichier avec le numéro spécifié n'a pas été ouvert.

154 Fichier déjà ouvert. Fermez-le d'abord et rouvrez-le dans le nouveau mode.

155 Nom d'unité de disque illégal, ou unité de disque non connectée.

156 Spécification de paramètre incorrecte dans l'instruction SET.

158 Commande non prise en charge. Commande illégale pour l'unité de disque.

159 Tentative d'écrire sur un disque protège en écriture.

160 Disque non inséré dans l'unité de disque spécifiée.

161 Le disque n'a pas été mis en forme par l'instruction INIT.

Codes des erreurs	Description
162	Erreur écriture-lecture de disque.
163	Mauvais disque dans l'unité. Les disques ont été changés quand un fichier était ouvert.
164	Erreur de disque plein.
165	Fin de fichier atteinte avec l'instruction INPUT#. Toutes les données ont été lues.
166	Espace mémoire insuffisant pour la zone de travail interne du disque. Pas assez d'espace libre dans la zone mémoire du PC-1600 pour le système E/S de l'unité de disque.
167	Erreur fatale de disque. Le contenu est détruit ou endommagé.
168	L'unité de disque CE-1600F ne fonctionne pas correctement. Défaut du matériel ou épuisement des piles.

Annexe G

Liste des Commandes BASIC

Nom de la commande	Description
ABS	Renvoie la valeur absolue de X
ACS	Renvoie la valeur de l'arc cosinus de X
ADIN ON/OFF/STOP	Valide/invalide les interruptions analogiques
AIN	Renvoie le niveau d'entrée analogique
ALARM$	Spécifie la durée d'alarme et le message
AREAD	Lit une variable de l'écran
ARUN	Commence automatiquement l'exécution d'un programme
ASC	Renvoie le code caractère pour une chaîne
ASN	Renvoie l'arc sinus de X
ATN	Renvoie l'arc de tangente de X
AUTO	Commute le numérotage automatique des lignes
BEEP	Génère un bruit par le haut-parleur interne
BEEP ON/OFF	Valide/invalide la génération du son
BLOAD	Charge un programme de langage machine à partir du disque ou de la bande
BREAK ON/OFF	Valide/invalide la touche BREAK
BSAVE	Sauvegarde un programme de langage machine sur un disque ou une bande
CALL	Appelle un programme langage machine
CHAIN	Charge et exécute un programme-BASIC sur une bande
CHR$	Renvoie un caractère de son code ASCII
CLEAR	Efface toutes les variables en mémoire
CLOAD	Charge un programme BASIC à partir d'une bande (Mode PC-1500)
CLOAD?	Vérifie le chargement à partir de la bande (Mode PC-1500)
CLOADM	Charge un programme langagemachine à partir d'une bande. (Mode PC-1500)
CLOSE	Ferme un fichier périphérique
CLS	Nettoie l'écran d'affichage
COLOR	Spécifié la couleur du stylo pour l'imprimante
COM$	Renvoie les paramètres de transmission
COMn ON/OFF/STOP	Valide/invalide les interruptions de transmission
CONT	Reprend l'exécution après STOP ou BREAK
COPY	Copie un fichier sur le disque ou la bande
COS	Renvoie le cosinus de X
CSAVE	Sauvegarde un programme BASIC sur bande (Mode PC-1500)

Nom de la commande	Description
CSAVEM	Sauvegarde un programme langage machine sur bande (Mode PC-1500)
CSIZE	Spécifié la taille des caractères à l'imprimante
CURSOR	Positionne le curseur sur l'écran
DATA	Liste les items de données pour l'instruction READ
DATES	Renvoie la date
DEG	Convertit les degrés en décimaux
DEGREE	Spécifié l'ordinateur en mode DEGREE
DELETE	Annule les lignes d'un programme
DIM	Réserve l'espace mémoire pour les variables/tableaux
DMS	Convertit les degrés décimaux en deg/mn/sec
DSKF	Renvoie l'espace libre sur un périphérique de disque
END	Arrête l'exécution du programme en cours
EOF	Renvoie la valeur montrant la fin du fichier
ERASE	Efface les variables et tableaux spécifiés
ERL	Renvoie le numéro de ligne d'une erreur
ERN	Renvoie le numéro de code d'erreur d'une erreur
EXP	Renvoie l'exposant élevé à la puissance X
FILES	Affiche le répertoire d'un disque
FOR..NEXT	Permet l'exécution réitérée des lignes de programme
GCURSOR	Positionne le curseur pour les graphiques sur l'écran
GLCURSOR	Positionne le stylo de l'imprimante dans le mode graphiques
GOSUB..RETURN	Branche sur un sous-programme
GOTO	Branchement inconditionnel à un numéro de ligne
GPRINT	Dessine des graphiques en images binaires sur l'écran d'affichage
GRAD	Spécifie l'ordinateur en mode GRADIENT
GRAPH	Spécifie l'imprimante en mode graphique
HEX$	Renvoie la chaîne hexadécimale pour un numéro
IF..THEN	Branchement conditionnel
INIT	Initialise un module, un disque; spécifie le tampon récepteur
INKEY$	Lit un caractère à partir du tampon du clavier
INP	Renvoie la donnée du port du microprocesseur
INPUT	Introduit les données du clavier dans un programme
INPUT$	Lit les enregistrements d'un fichier
INSTAT	Renvoie les états du signal de commande pour le port série
INSTR	Recherche un caractère dans la chaîne
INT	Tronque la partie décimale d'un nombre
KBUFF$	Ecrit les caractères dans le tampon du clavier
KEY ON/OFF/STOP	Valide/invalide les touches de fonction

Nom de la commande	Description
KEYSTAT	Spécifie la répétition de touche, les fonctions de cliquetage des touches
KILL	Efface un fichier sur le disque
LCURSOR	Déplace le stylo de l'imprimante à une position spécifiée
LEFT$	Renvoie les caractères de l'extrémité gauche d'une chaîne
LEN	Renvoie le nombre de caractères dans une chaîne
LET	Assigne une valeur à une variable
LF	Alimente le papier dans l'imprimante
LFILES	Imprime le répertoire sur sortie papier
UNE	Tire un trait entre des points sur l'écran
LIST	Liste un programme sur l'écran
LUNE	Tire un trait entre des points sur sortie papier
LN	Renvoie le logarithme naturel de X
LOAD	Charge un fichier du disque en mémoire
LOC	Renvoie un numéro d'enregistrement sollicité dans un fichier
LOCK/UNLOCK	Valide/invalide la touche MODE
LOF	Renvoie la capacité d'un fichier sur le disque
LOG	Renvoie le logarithme commun (base 10) de X
LPRINT	Sort la donnée sur l'imprimante ou le port série
MAXFILES	Spécifie le nombre maximal de fichiers pour un programme
MEM	Renvoie l'espace mémoire inutilisé dans la zone utilisateur
MERGE	Classe un programme sur bande en mémoire
MID$	Renvoie une chaîne de l'intérieur d'une autre chaîne
MOD	Renvoie le reste d'une division
MODE	Sélectionne le mode écran pour la compatibilité du PC-1500
NAME	Renomme un fichier sur le disque
NEW	Vide la mémoire/assigne un espace pour le code machine
ON ADIN GOSUB	Branche sur une interruption analogigue
ON COMn GOSUB	Branche sur une interruption en port série
ON ERROR GOTO	Branche sur à un programme de traitement des erreurs sur une erreur
ON..GOSUB/ON..GOTO	Branchement conditionnel multiple
ON KEY GOSUB ON	Branche sur l'introduction d'une touche de fonction
PHONE GOSUB ON	Branche sur l'introduction d'un modem téléphone
TIMES GOSUB OPEN	Branche à une durée spécifiée
OUT	Ouvre un fichier pour l'accès
	Ecrit une donnée dans l'accès du microprocesseur

Nom de la commande	Description
OUTSTAT	Spécifie les états du signal de commande pour le port, série
PAPER	Spécifie le type de papier et la gamme d'impression verticale
PASS	Valide/invalide le mot de passe
PAUSE	Affiche les données sur l'écran pour une durée fixe
PCONSOLE	Spécifie le format d'impression/le code EOL pour l'imprimante/les ports
PEEK	Renvoie l'octet de donnée de la mémoire (mode PC-1600)
PHONE ON/OFF/STOP	Valide/invalide les interruptions RS-232C
PITCH	Spécifié l'espacement des caractères/l'interlignage pour l'imprimante
POINT	Renvoie la valeur des commandes par points à un endroit sur l'écran
POKE	Ecrit un octet de donnée en mémoire (mode PC-1600)
POWER	Spécifie la mise hors tension automatique
PRESET	Réinitialise une commande par point à un endroit sur l'écran
PRINT	Sort une donnée sur l'écran
PRINT#	Ecrit une donnée sur un fichier
PSET	Spécifie/respécifie une commande par point sur l'écran
PZONE	Spécifie une zone d'impression pour l'imprimante ou le port série
RADIAN	Spécifie l'ordinateur en mode RADIAN
RANDOM	Initialise la génération de nombres aléatoires
RCVSTAT	Spécifie le protocole de réception/le dépassement pour le port série
READ..DATA	Lit les données des lignes DATA dans un programme
REM	Utilisée pour inclure un commentaire dans un programme
RENUM	Numérote à nouveau les lignes d'un programme
RESTORE	Utilisée pour relire les données des lignes DATA
RESUME	Reprend l'exécution d'un programme après une erreur de programme
RETI	Renvoie d'un sous-programme d'interruption
RIGHT$	Renvoie les caractères de l'extrémité droite d'une chaîne
RLINE	Tire une ligne en coordonnées relatives sur l'imprimante
RMT ON/OFF	Valide/invalide la télécommande de bande
RND	Génère un nombre aléatoire
ROTATE	Spécifie l'orientation de l'impression et le déplacement de la tête imprimante.
RUN	Lance l'exécution d'un programme
RXD$	Renvoie les données réelles du port série

Nom de la commande	Description
SAVE	Sauvegarde un programme BASIC sur un disque
SET	Spécifié la protection en écriture sur le fichier d'un disque
SETCOM	Spécifie le protocole de transmission pour les ports série
SETDEV	Sélectionne un port série pour la sortie
SGN	Renvoie le signe d'une expression X
SIN	Renvoie le sinus d'une expression X
SNDBRK	Envoie les caractères de rupture à un port série
SNDSTAT	Spécifie le protocole d'émission/le dépassement pour le port série
SORGN	Spécifie la position du stylo actuel en origine
SQR	Renvoie la racine carrée de X
STATUS	Renvoie la capacité de l'espace libre pour les zones mémoire
STOP	Arrête l'exécution durant la mise au point d'un programme
STR$	Convertit les données numériques en données alphanumériques
TAB	Déplace le stylo de l'imprimante à une colonne spécifique
TAN	Renvoie la tangente de X
TEST	Lance un essai sur l'imprimante
TEXT	Spécifie l'imprimante en mode TEXT
TIME	Spécifie/renvoie l'heure de l'horloge incorporée
TIMÈ$	Renvoie l'heure de l'horloge incorporée sous forme d'une chaîne alphanumérique
TIME$ ON/OFF/STOP	Valide/invalide les interruption d'horloge
TITLE	Sélectionne la zone mémoire de l'ordinateur
TRON/TROFF	Spécifie/annule les traces de mise au point d'un programme
VAL	Convertit des chaînes numériques en valeurs décimales
WAIT	Spécifie la durée d'attente après l'instruction PRINT
WAKE$	Spécifie la chaîne alphanumérique heure/commande pour la mise sous tension automatique
XCALL	Appelle un programme langage machine (mode PC-1500)
XPEEK/XPEEK#	Renvoie un octet de donnée de la mémoire (mode PC-1500)
XPOKE/XPOKE#	Ecrit un octet de donnée en mémoire (mode PC-1500)

Annexe H

Compatibilité avec le PC-1500 et ses PERIPHERIQUES

Les programmes écrits en BASIC sur le PC-1500 peuvent en général être lancés sur le PC-1600 en MODE 1. La plus grande partie des noms des commandes BASIC sont identiques entre les deux modèles, mais il y a des différences dans les valeurs par défaut pour la largeur du papier de l'imprimante, qui doivent être rectifiées dans les programmes qui sortent sur imprimante. Les commandes du PC-1600 qui sont associées avec un disque et des fichiers RAM sur disques, ainsi que les commandes d'interruption en port série ne sont pas prises en charge sur le PC-1500. Les programmes du PC-1500 stockés sur bandes peuvent être chargés sur le PC-1600. Cependant, vous devez garder quelques points à l'esprit, particulièrement en ce qui concerne les commandes qui accèdent directement à la mémoire. Les commandes qui suivent sont différentes entre les deux modèles:

Equivalences entre les commandes

PC-1600	PC-1500
TAB	LCURSOR.
LLINE	UNE
UNE	pas d'équivalence
XCALL	CALL
CALL	pas d'équivalence
XPOKE	POKE
POKE	pas d'équivalence
XPEEK	PEEK
PEEK	pas d'équivalence
XPOKE#	POKE#
XPEEK#	PEEK#

Note: Reportez-vous à l'Annexe E pour les détails sur les commandes concernant le langage machine et leur compatibilité.

Il n'est pas possible de spécifier TIME=0 sur le PC-1600 bien que cette instruction soit couramment utilisée dans les programmes du PC-1500 pour réinitialiser le compteur de temporisation.

Emulation du PC-1500

Quand le PC-1600 est spécifié en MODE 1 par la commande MODE,'il se comporte comme le PC-1500; seule la ligne du bas de l'écran est active et l'écran est commandé comme pour le PC-1500. Le jeu de caractères est également différent entre le jeu de caractères du PC-1600 et celui du PC-1500; il n'y a pas de caractères graphiques, internationaux ou grecs disponibles pour les codes &80 et &FF et les symboles correspondants aux codes &27 et &5B changent. Voyez l'Annexe C pour les tableaux des codes de caractères.

Introduction d'un programme BASIC du PC-1500 dans le PC-1600

1Tapez le programme en mode PRO comme d'habitude, en vous assurant que la commande UNE est bien changée en commande LLINE à chaque fois que vous procédez à cette opération.

2Sauvegardez le programme sur un équipement périphérique, tout en ajoutant par prudence, une ligne de commentaires en tête du programme pour vous souvenir qu'il s'agit d'un programme du PC-1500.

Programmes stockés sur cassettes

Les programmes du PC-1500 stockés sur cassette via l'imprimante CE-150/ interface à cassette du PC-1500 peuvent être chargés dans le PC-1600 en MODE 1 sans modification avec la commande CLOAD.

Cependant, les programmes du PC-1600 stockés sur cassette par l'imprimante CE-1600P avec unité interface de cassette, ne peuvent être chargés dans le PC-1500. Ceci, parce que le format de bande est différent sous certains aspects et que l'allocation mémoire des deux ordinateurs est différente.

Lancement d'un programme BASIC PC-1500 sur le PC-1600

1Assurez-vous qu'il y a un module dans le logement 1 ou le logement 2. Ceci pour vous assurer que l'espace mémoire est suffisant. Utilisez l'un des types suivants: CE-151, CE-155, CE-159, CE-161 et la capacité doit être inférieure à 16K octets. S'il n'y a pas de module dans aucun des logements, un code d'erreur sera alors affiché.

2Spécifiez le PC-1600 en MODE 1 avec la commande MODE

3 Lancez le programme de la manière habituelle. Les résultats seront affichés sur l'écran exactement comme sur le PC-1500.

Sortie imprimée sur l'imprimante CE-1600P

Toutes les commandes de l'imprimante du PC-1500 fonctionnent sur l'imprimante du PC-1600, mais du fait de la différence dans la largeur du papier entre les imprimantes CE-1600P et CE-150, si vous voulez que la largeur de la sortie papier soit identique à celle du CE-150, vous devez spécifier la largeur du papier. Tapez les commandes suivantes en mode direct ou ajoutez-les en haut de votre programme:

```
CSIZE 2
PZONE "LPT1:",18
PCONSOLE "LPT1:",18,0,0
```

Les programmes stockés sur bande utilisant le PC-1500 avec l'unité interface CE-150 peuvent être chargés sur le PC-1600 par l'unité interface CE-1600P et pourront être exécutés sans problème. Mais, si le programme original contient les instructions LCURSOR, lors de la sortie du listage sur la CE-1600P, LCURSOR apparaîtra sous le symbole ~. Pour contourner ce problème, vous pouvez éditer le programme sur le PC-1600 en remplaçant l'instruction LCURSOR par la fonction TAB.

Utilisation du PC-1600 avec l'imprimante CE-150/înterface à cassette

Le PC-1600 peut être monté sur l'imprimante CE-150/interface à cassette ou avec les unités d'interface CE-158 et CE-162E du PC-1500. Spécifié en MODE 1, l'ordinateur fonctionnera exactement comme le PC-1500. Cependant, quand les programmes du PC-1600 sont listés via ces périphériques, les noms des commandes du PC-1600 seront imprimés dans leurs équivalents du PC-1500. Ceci parce que le logiciel interne incorporé dans ces unités ne contient que les noms des commandes du PC-1500. Ceci signifie, par exemple, que les commandes LLIST dans un programme seront imprimées sur le listage du programme en tant que LIST. Lors de l'impression sur l'imprimante CE-150, l'interrupteur PRINT doit être positionné sur le côte •. Les calculs directs ne seront pas imprimés.

Compatibilité du PC-1500 A

La zone dans la RAM interne (7C00 à 7FFF) qui est la zone libre de l'utilisateur dans le PC-1500A n'est pas disponible pour l'utilisateur du PC-1600; elle est utilisée par le système. Pour cette raison, le PC-1500A n'est pas compatible avec le PC-1600.

Compatibilité avec des modèles plus anciens du PC-1500

Dans des modèles plus anciens du PC-1500, la valeur maintenue dans le compteur FOR..NEXT donne un résultat 1 supérieur à celui obtenu sur le PC-1600, après l'exécution d'une boucle FOR..NEXT. Par exemple la boucle suivante:

```
10 FOR K=1 TO 10
20 NEXT K 30
PRINT K
```

La valeur de K imprimé sera de 11 sur le PC-1600 mais de 10 sur les modèles plus anciens du PC-1500.

De même, dans l'exemple suivant:

```
10 S=0
20 FOR K=1 TO 10 STEP 4
30 S=S+1
40 NEXT K
50 PRINT S
```

La valeur de S sera de 3 sur le PC-1600 et de 4 sur le modèle plus ancien du PC-1500.

Le modèle plus ancien du PC-1500 évalue également l'instruction IF..THEN différemment. La valeur de l'expression dans l'instruction IF..THEN est évaluée comme TRUE (vraie) ou FALSE (fausse)

MODELE •	Evaluée comme	
	TRUE	FALSE
PC-1600 PC-1500	*0	0
Ancien modèle PC-1500	>0	<0

Pour déterminer si un PC-1500 est l'un de ces anciens modèles, regardez dans l'adresse &C5C0 à l'aide de la commande PEEK: PEEK&C5C0. Si la valeur renvoyée est 6, vous avez alors un modèle ancien du PC-1500. Il n'y a donc aucun problème pour les boucles FOR..NEXT.

Annexe I

Entretien et Pannes

Votre ordinateur de poche SHARP PC-1600 est un équipement de précision qui nécessite constamment les plus grands soins. Tout au long de ce manuel, nous vous avons fourni un certain nombre de notes et d'avertissements, anticipant les points où vous pouviez éprouver des difficultés. Ces points couvrent une grande plage de sujets durant l'exploitation, la manipulation et la programmation.

Cette section vous offre quelques conseils pour conserver votre PC-1600 en bon état de fonctionnement — et que faire dans le cas contraire.

Voici la liste des points clés sur l'entretien général de votre PC-1600.

1 Affichage L'écran LCD de votr PC-1600 est en verre. Il est fragile et peut se casser si vous n'en prenez pas soin. Lorsque vous ne l'utilisez pas, conservez votre ordinateur dans son étui souple protecteur. Faites particulièrement attention à ne pas heurter l'écran avec les connecteurs des câbles et les accessoires, quand vous utilisez les équipements périphériques.

2Endroits de rangement et de travail Evitez les endroits aux températures et à l'humidité extrêmes. La chaleur engendrée par une exposition prolongée sous les rayons du soleil peut endommager sévèrement votre ordinateur. L'humidité sous toutes ses formes est également une ennemie. Quand le temps est sec, faites attention à l'électricité statique, accumulée dans les moquettes, qui peut détruire la mémoire.

3Nettoyage Utilisez seulement un chiffon sec et doux pour nettoyer l'ordinateur et ses périphériques. N'utilisez jamais de savon, de produits de nettoyage commerciaux ou tout autre liquide.

4Fuite des piles Enlevez toujours les piles avant de ranger pendant un long moment votre ordinateur, afin d'éviter les dommages provoqués par la fuite des piles et les réactions chimiques.

5Réparations Si votre PC-1600 a besoin d'être réparé, contactez votre concessionnaire SHARP ou un centre de réparation. Une réparation effectuée par une personne non compétente ou une altération de votre ordinateur peuvent provoquer des problèmes supplémentaires.

Votre PC-1600 répond de différentes manières pour éviter les problèmes ou vous alerter d'érentuels problèmes par les codes ERROR, les messages CHECK, la mise hors tension automatique, la sauvegarde mémoire et l'avertissement des piles épuisées. Consultez ce manuel et les manuels des équipements en option que vous utilisez, si des problèmes surgissaient.

La liste de vérification suivante est un guide qui vous permettra de détecter les problèmes les plus courants. Voyez également le chapitre 13 sur la Mise au point du langage BASIC lorsque vous avez une difficulté quelconque avec les programmes.

Nous suggérons que vous essayiez de résoudre vous-même vos problèmes et ne consultiez votre concessionnaire qu'en dernier recours. Vous pouvez avoir provoqué le problème par inadvertance et le résoudre vous-même est une autre manière d'en apprendre plus sur votre SHARP PC-1600.

Conseils en cas de pannes

Si ceci se produit...	Faites ceci...
L'ordinateur est sous tension mais l'écran est blanc	1. Pressez les touches OFF et ON à nouveau pour vérifier que l'appareil est vraiment sous tension 2.Vérifiez le témoin de l'état des piles 3.Si vous utilisez l'adaptateur CA, vérifiez qu'il est bien branché 4.Réglez la molette de commande du contraste.
L'affichage est bien mais les touches sont inopérables	1. Pressez CL pour remettre à zéro 2.Mettez l'ordinateur hors et sous tension à nouveau 3.Pressez la touche RESET seule pour une réinitialisation simple 4.Pressez la touche ON et la touche RESET pour ALL RESET (réinitialisatior générale)

Les calculs introduits sont affichés en format d'instruction BASIC avec les deux points (:) après le premier nombre

Pressez la touche MODE pour passer du mode PROgramme au mode RUN pour les calculs directs.

AnnexeJ

Spécifications

Microprocesseurs	
UC principale	SC7852 (Z-80A CMOS compatible; 3,58 MHz)
Unité esclave	LH5803 (compatible PC-1500; 1,3 MHz)
Sous UC	LU57813P (307,2 kHz)
Affichage Caractères alphanumériques	A cristaux liquides; Contraste réglable 26 colonnes x 4 lignes 156 x 32 points
Graphiques	63 touches alphanumériques; 6 touches de fonction
Clavier	
Mémoire	
ROM	96K octets
RAM	16K octets (extensible jusqu'à 80K octets) Zone utilisateur —11834 octets
Extension de RAM	2 logements à l'arrière pour les modules d'extension
Interfaces	E/S série RS-232C, E/S séquentielle optique, entrée analogique
Caractéristiques	Affichage de l'épuisement des piles horloge en temps réel incorporée fonctions mise sous/hors tension automatique et alarme Interruptions externes prises en charge fonctions de transmissions
Alimentation	6V⎓-(CC) Quatre piles sèches (SUM-3,AA ou R6) Adaptateur CA (EA-160 ou EA-150)
Consommation	0,48W
Piles	25 heures à 20°C (68°F): 10mn d'exploitation, 50 mn d'affichage par heure
Dimensions	195(L) x 86(P) x 25,5(H) mm 7 11/16"(W) x 3 3/8"(D) x 1"(H)
Poids	390g env. (0,86 Ibs) (avec les piles)
Température	0°C à 40°C (32°F à 104°F)
Accessoires	Etui souple Deux grilles de clavier Quatre piles Manuel de fonctionnement